Chapter_3_Part_II

coop711

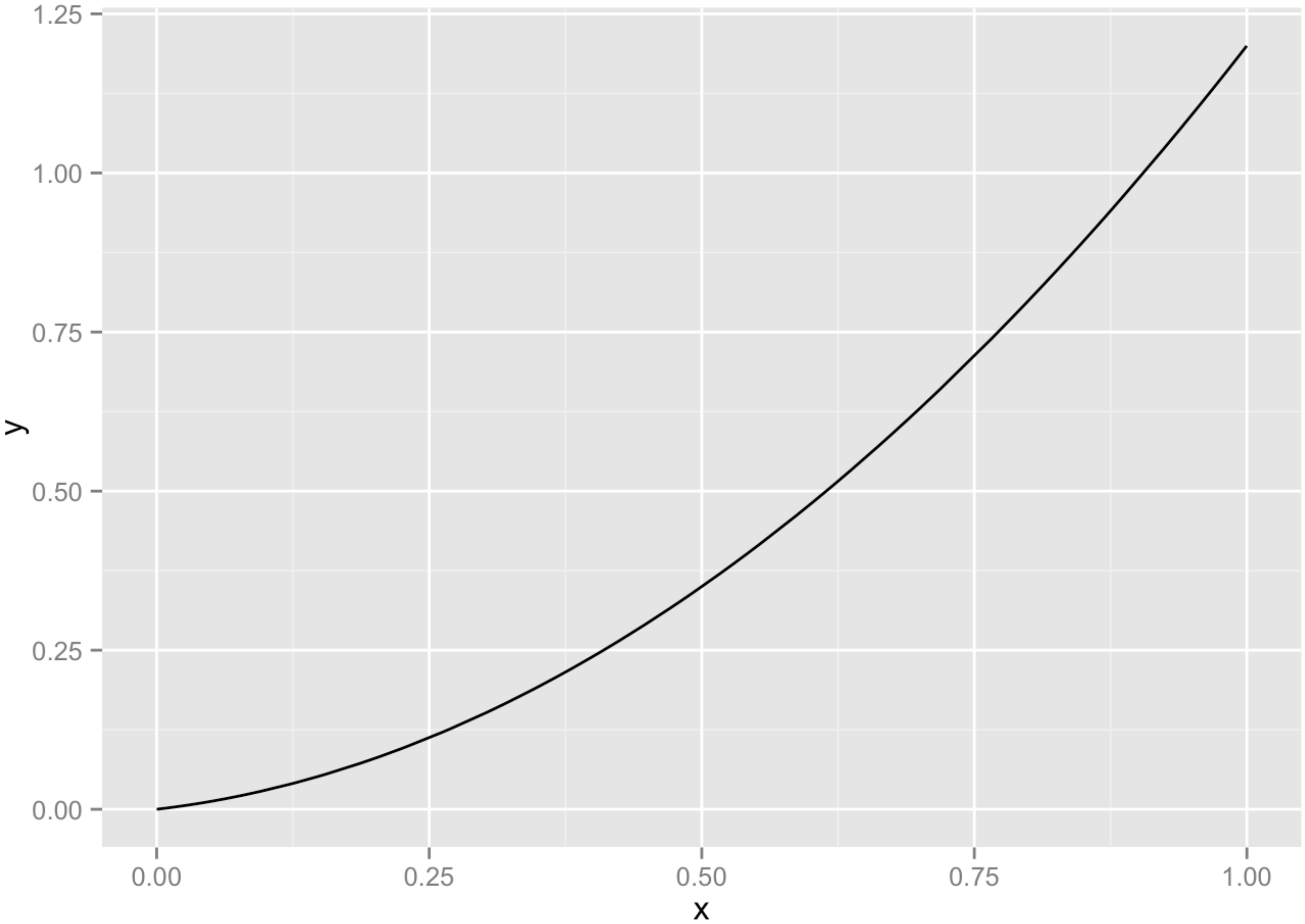
2015년 9월 12일

Relationship

Line Plots

- Listing 3.11

```
library(scales)
library(ggplot2)
x <- runif(100)
y <- x^2 + 0.2*x
ggplot(data.frame(x=x, y=y), aes(x=x, y=y)) + geom_line()
```



Scatter Plots and Smoothing Curves

- Listing 3.12

```
load("chapter_3_Part_I_0912.rda")
ls()
```

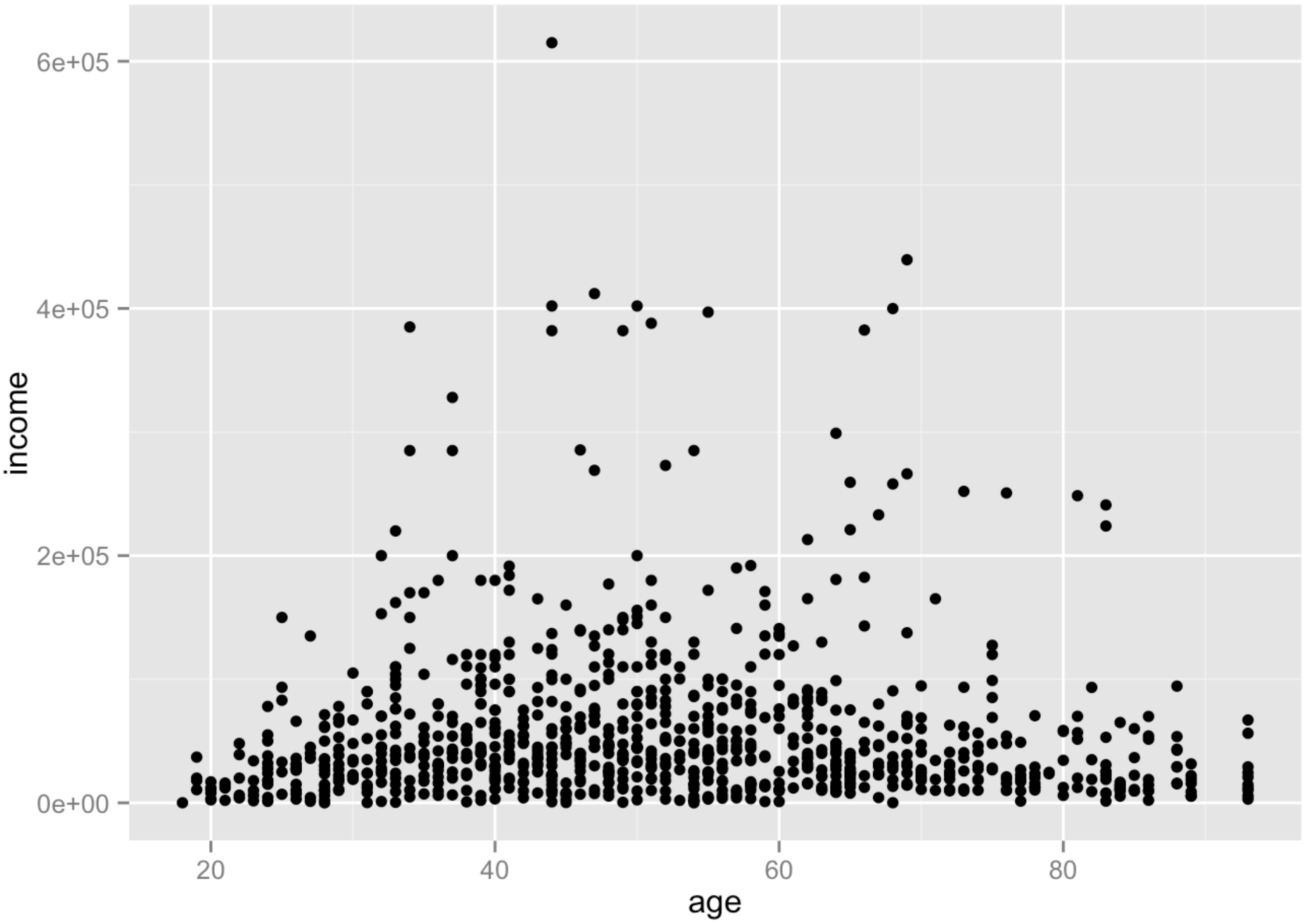
```
##  [1] "age.ecdf"  "cowmap"    "custdata"  "dhus"      "dpus"      "dtest"
##  [7] "dtrain"    "g.ecdf"    "g1"        "g2"        "g3"        "g4"
## [13] "g5"        "o.sor"     "p"         "p1"        "p2"        "p3"
## [19] "p4"        "poly.age"  "poly.x"    "poly.y"    "psub"      "result"
## [25] "schlmap"   "sor.df"    "sor.df.2"  "sor.df.o"  "sor.tbl"   "sub"
## [31] "theme.kr"  "x"         "y"
```

```
custdata2 <- subset(custdata, (custdata$age > 0 & custdata$age < 100 & custdat
a$income > 0))
options(digits=2)
cor(custdata2$age, custdata2$income)
```

```
## [1] -0.022
```

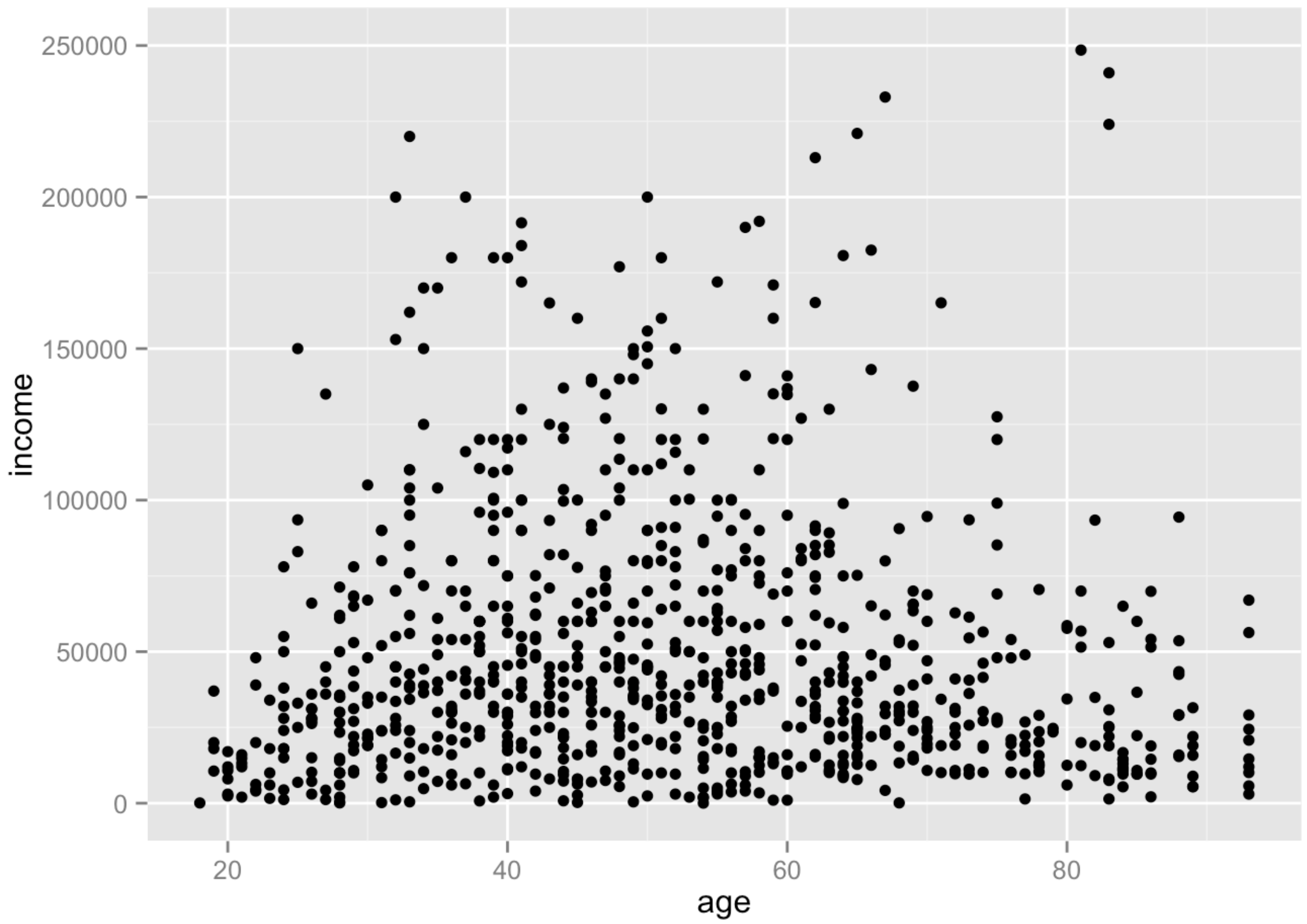
- Scatter Plot. 화살표를 넣기 위하여 `grid` 패키지 등록

```
library(grid)
(g1 <- ggplot(custdata2, aes(x=age, y=income)) + geom_point())
```



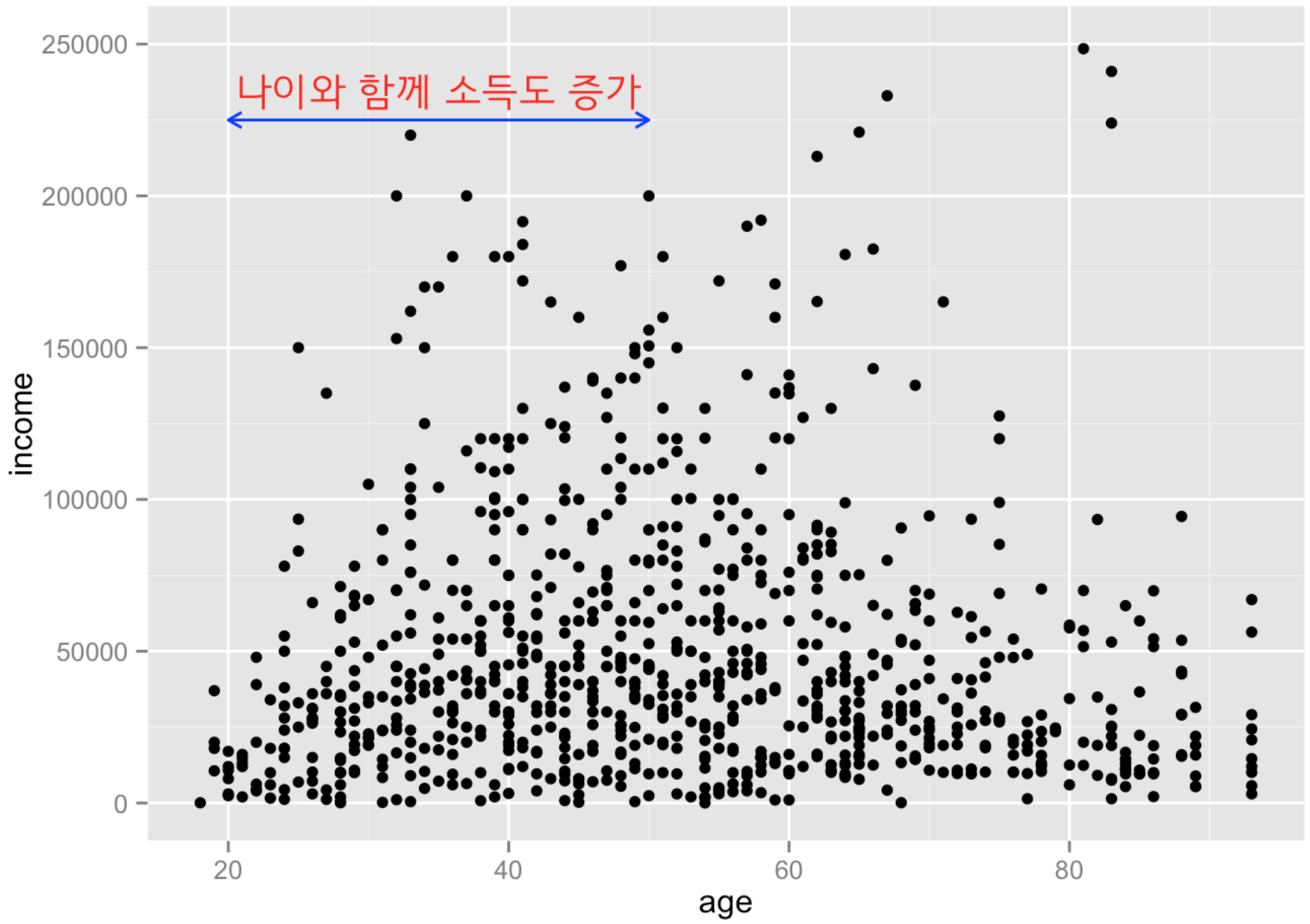
```
(g2 <- g1 + ylim(0, 250000))
```

```
## Warning: Removed 25 rows containing missing values (geom_point).
```



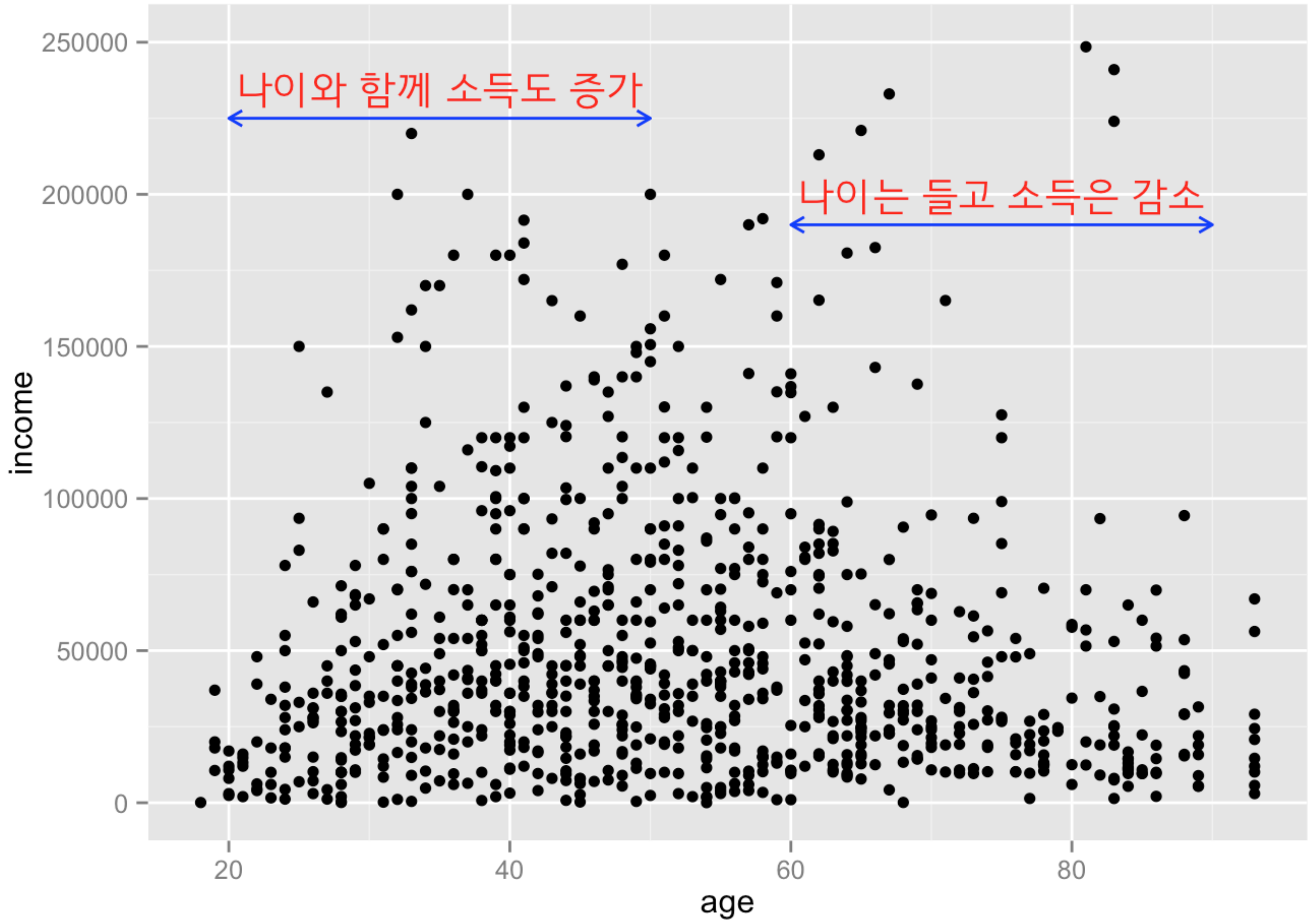
```
(g3 <- g2 + annotate("segment", x=20, xend=50, y=225000, yend=225000, colour="b
lue", size=0.5, arrow=arrow(ends="both", length=unit(0.2, "cm"))) +
   annotate("text", x=35, y=235000, label="나이와 함께 소득도 증가", family="HCR Dotu
m LVT", size=5, colour="red"))
```

```
## Warning: Removed 25 rows containing missing values (geom_point).
```



```
(g4 <- g3 + annotate("segment", x=60, xend=90, y=190000, yend=190000, colour="b
lue", size=0.5, arrow=arrow(ends="both", length=unit(0.2, "cm"))) +
   annotate("text", x=75, y=200000, label="나이는 들고 소득은 감소", family="HCR Dotu
m LVT", size=5, colour="red"))
```

```
## Warning: Removed 25 rows containing missing values (geom_point).
```

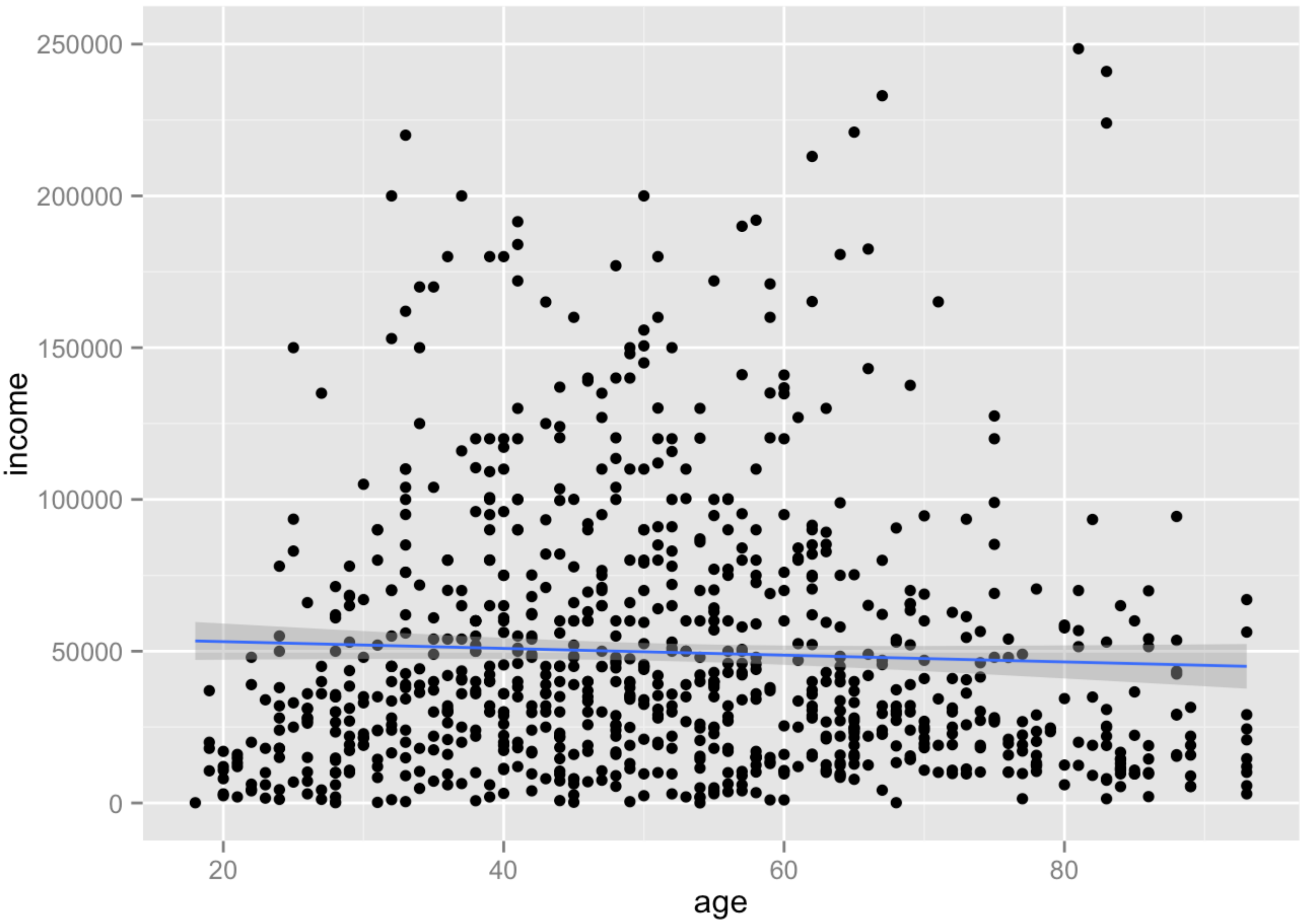


- Linear Fit 추가

```
g1 + stat_smooth(method="lm") + ylim(0, 250000)
```

```
## Warning: Removed 25 rows containing missing values (stat_smooth).
```

```
## Warning: Removed 25 rows containing missing values (geom_point).
```

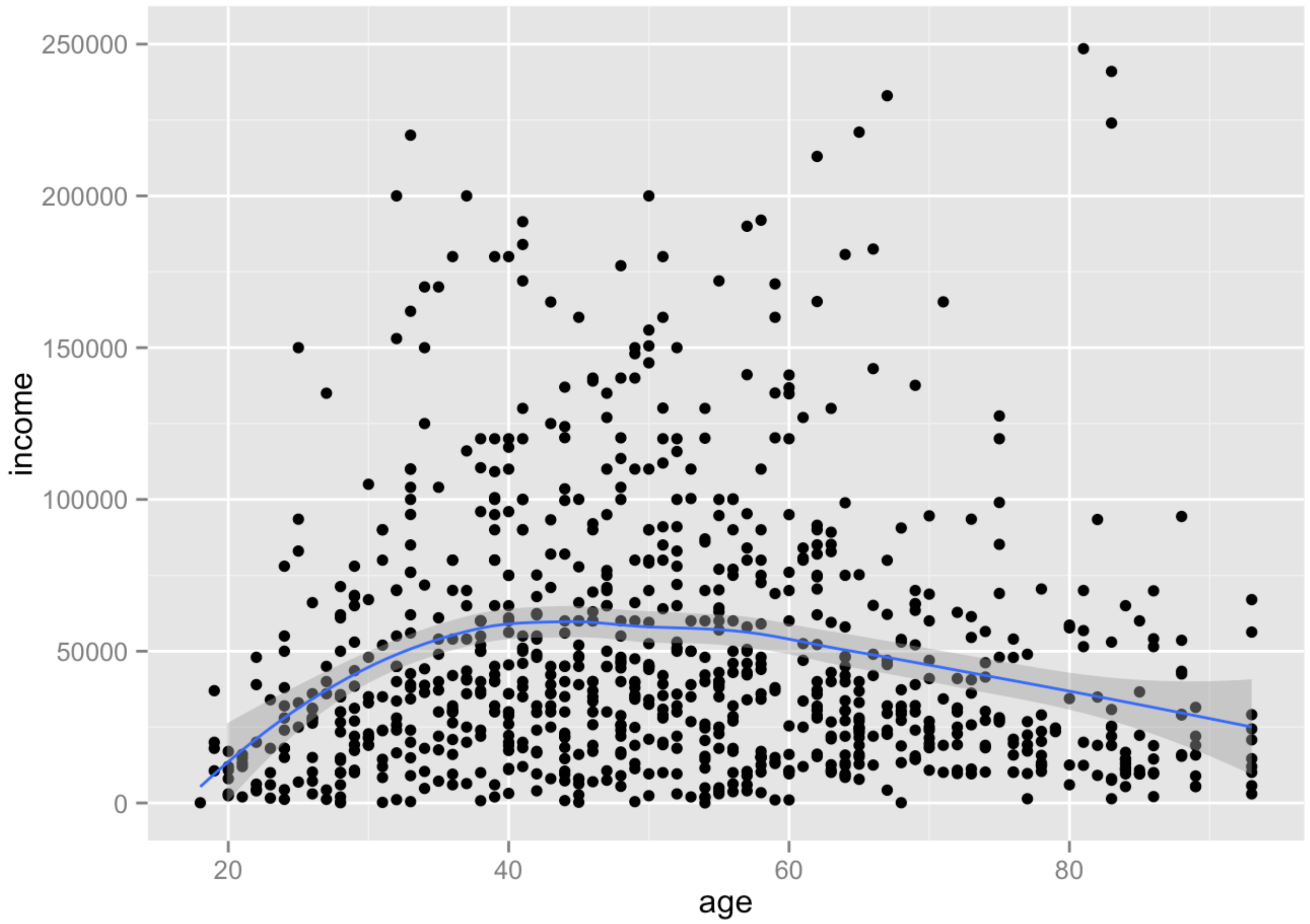


- local smoother 추가

```
g1 + stat_smooth(method="loess") + ylim(0, 250000)
```

```
## Warning: Removed 25 rows containing missing values (stat_smooth).
```

```
## Warning: Removed 25 rows containing missing values (geom_point).
```



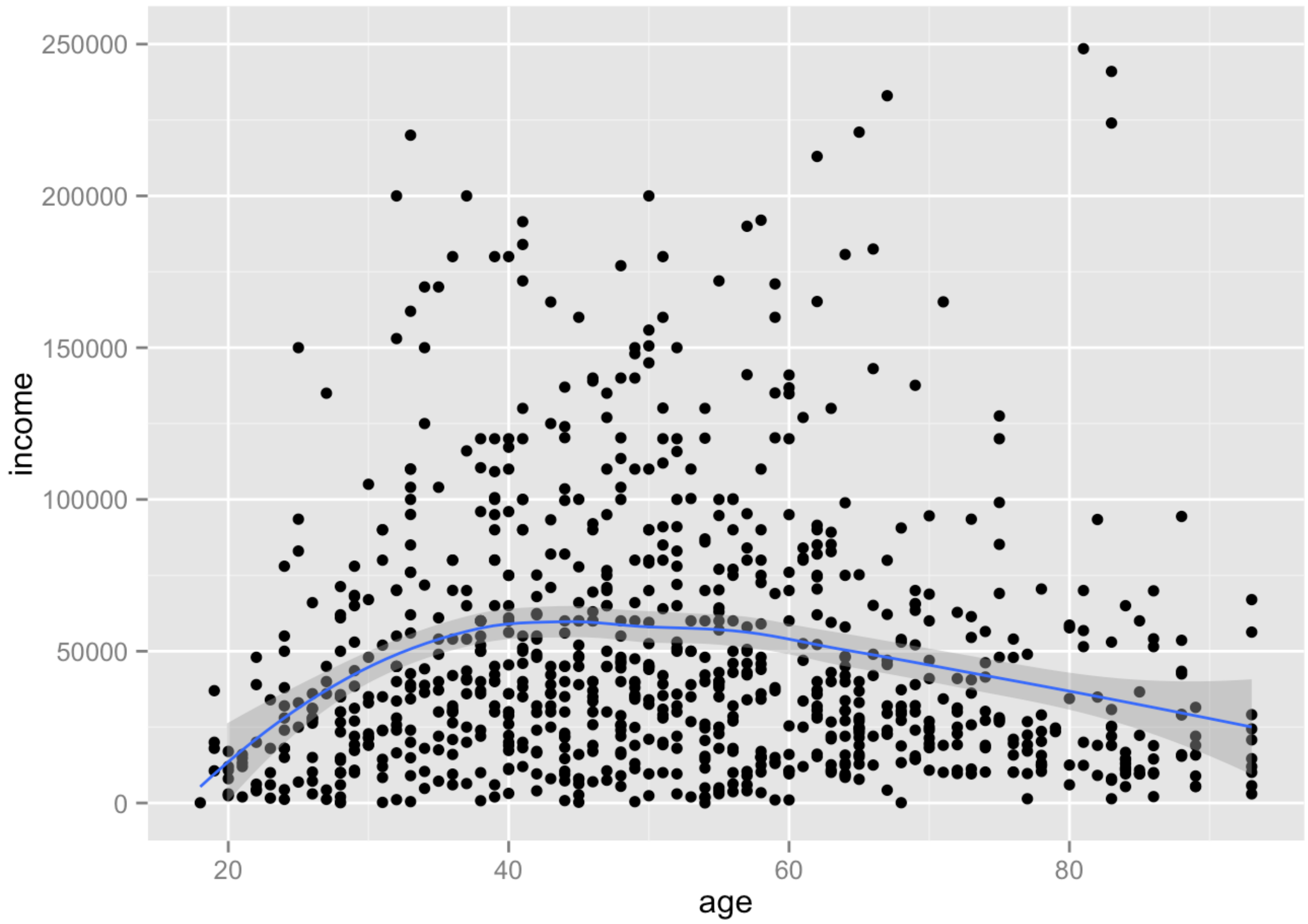
- geom_smooth() 로 하면,

```
g1 + geom_smooth() + ylim(0, 250000)
```

```
## geom_smooth: method="auto" and size of largest group is <1000, so using loes
s. Use 'method = x' to change the smoothing method.
```

```
## Warning: Removed 25 rows containing missing values (stat_smooth).
```

```
## Warning: Removed 25 rows containing missing values (geom_point).
```

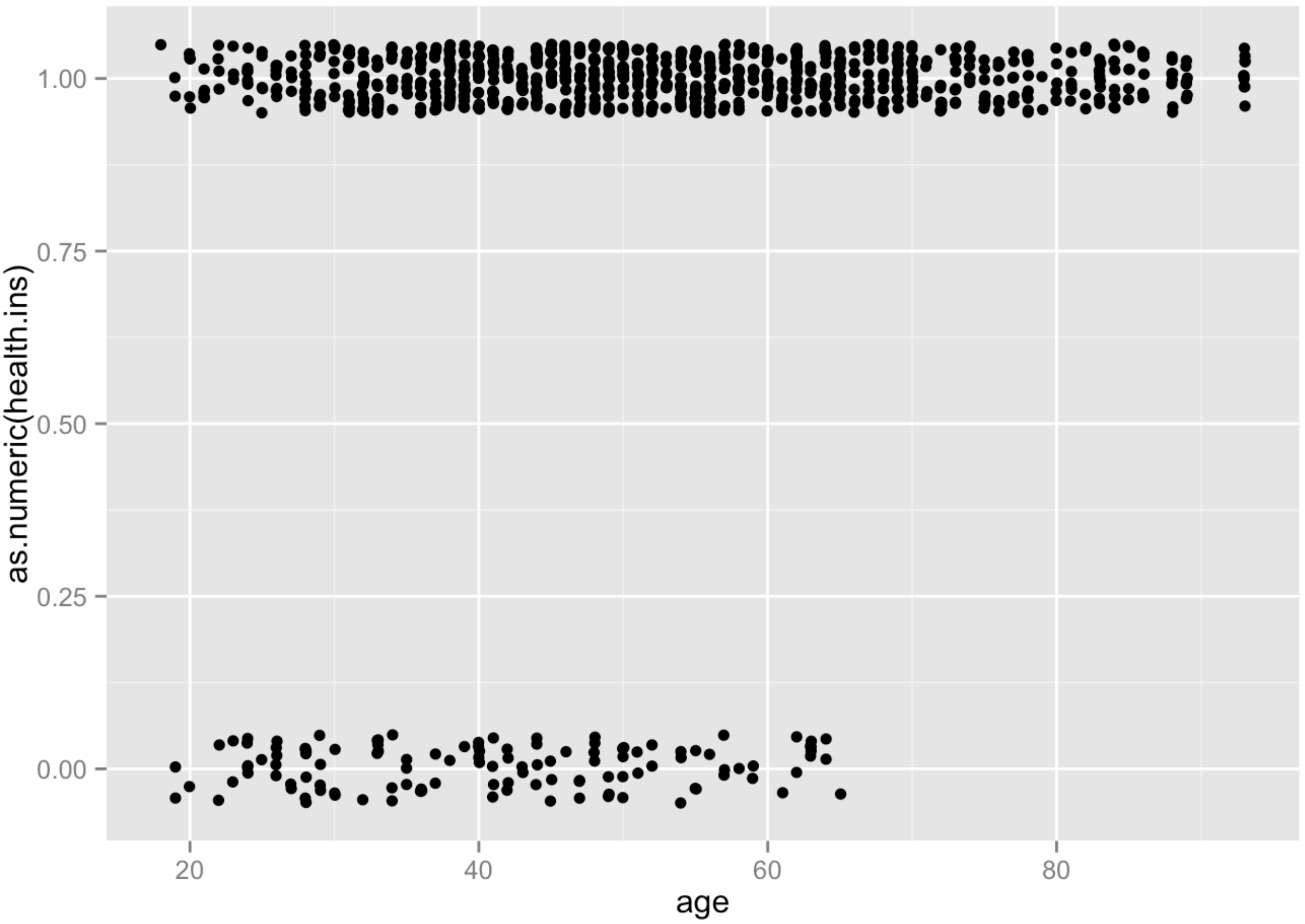


- Listing 3.13

```
summary(custdata2$health.ins)
```

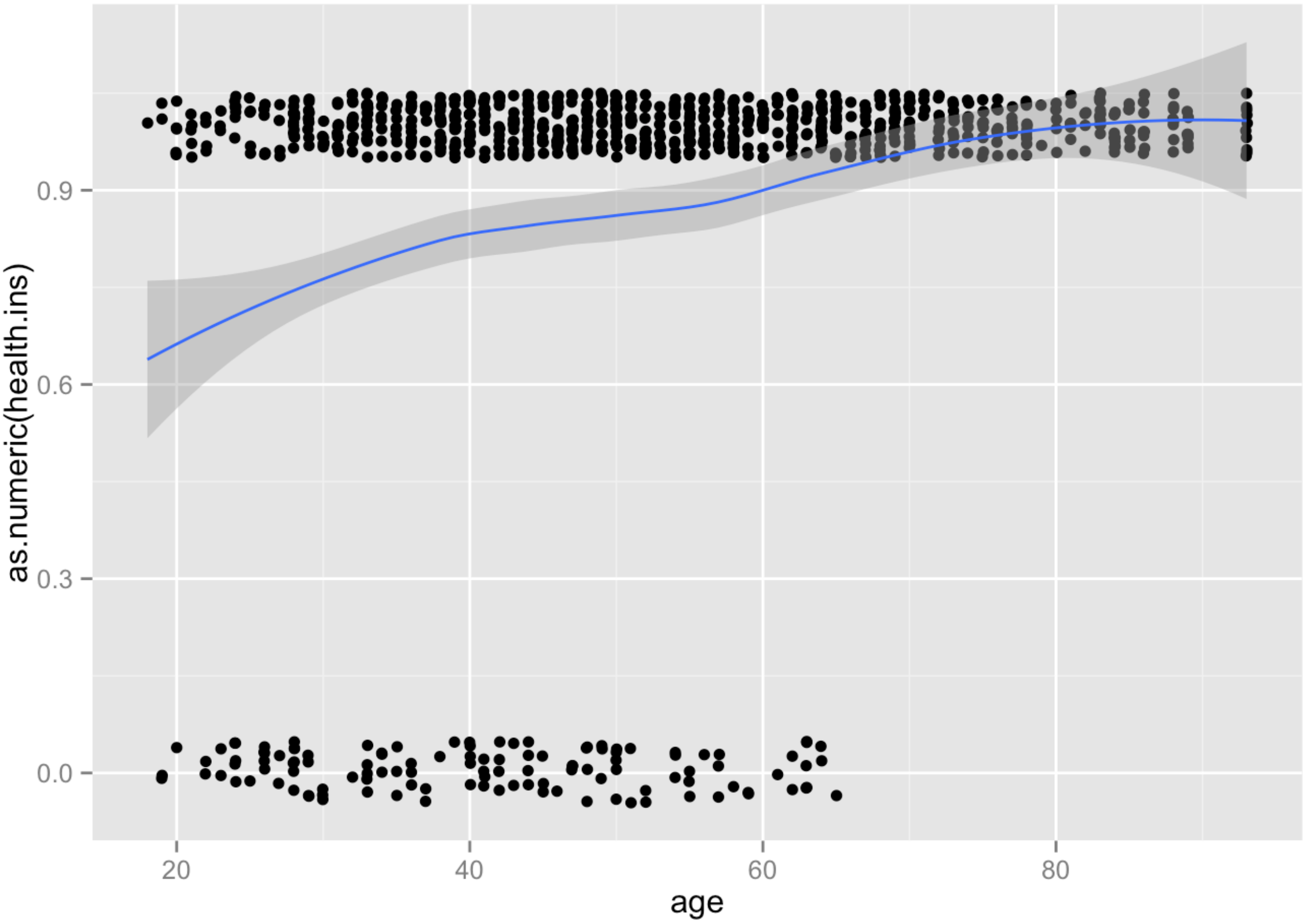
```
##    Mode   FALSE    TRUE    NA's
## logical     119     791       0
```

```
(h1 <- ggplot(custdata2, aes(x=age, y=as.numeric(health.ins))) +
  geom_point(position=position_jitter(w=0.05, h=0.05)))
```



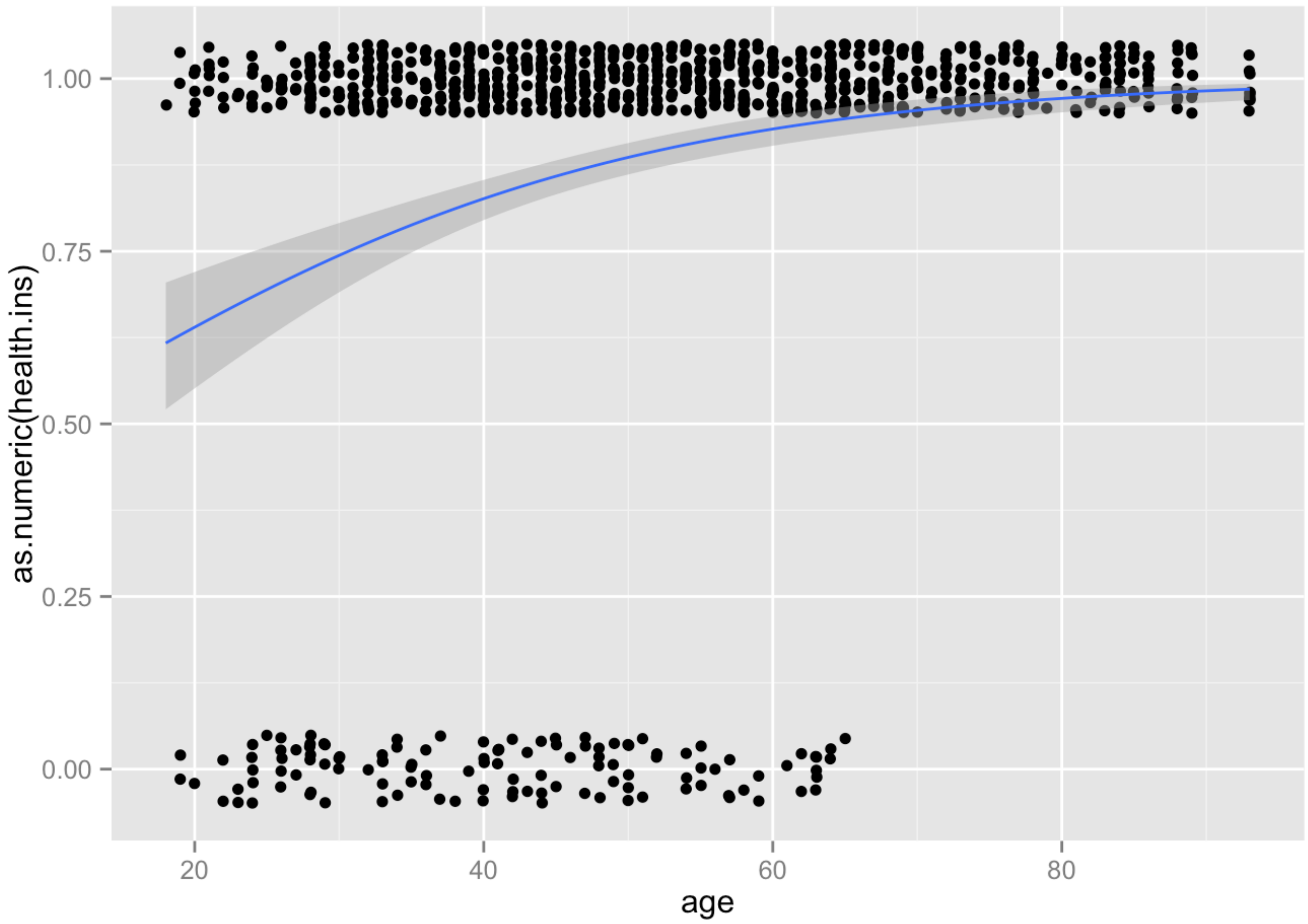
```
(h2 <- h1 + geom_smooth())
```

```
## geom_smooth: method="auto" and size of largest group is <1000, so using loes
s. Use 'method = x' to change the smoothing method.
```



- glm의 하나인 logistic regression으로 적합시키면,

```
(h3 <- h1 + stat_smooth(method=glm, family=binomial))
```



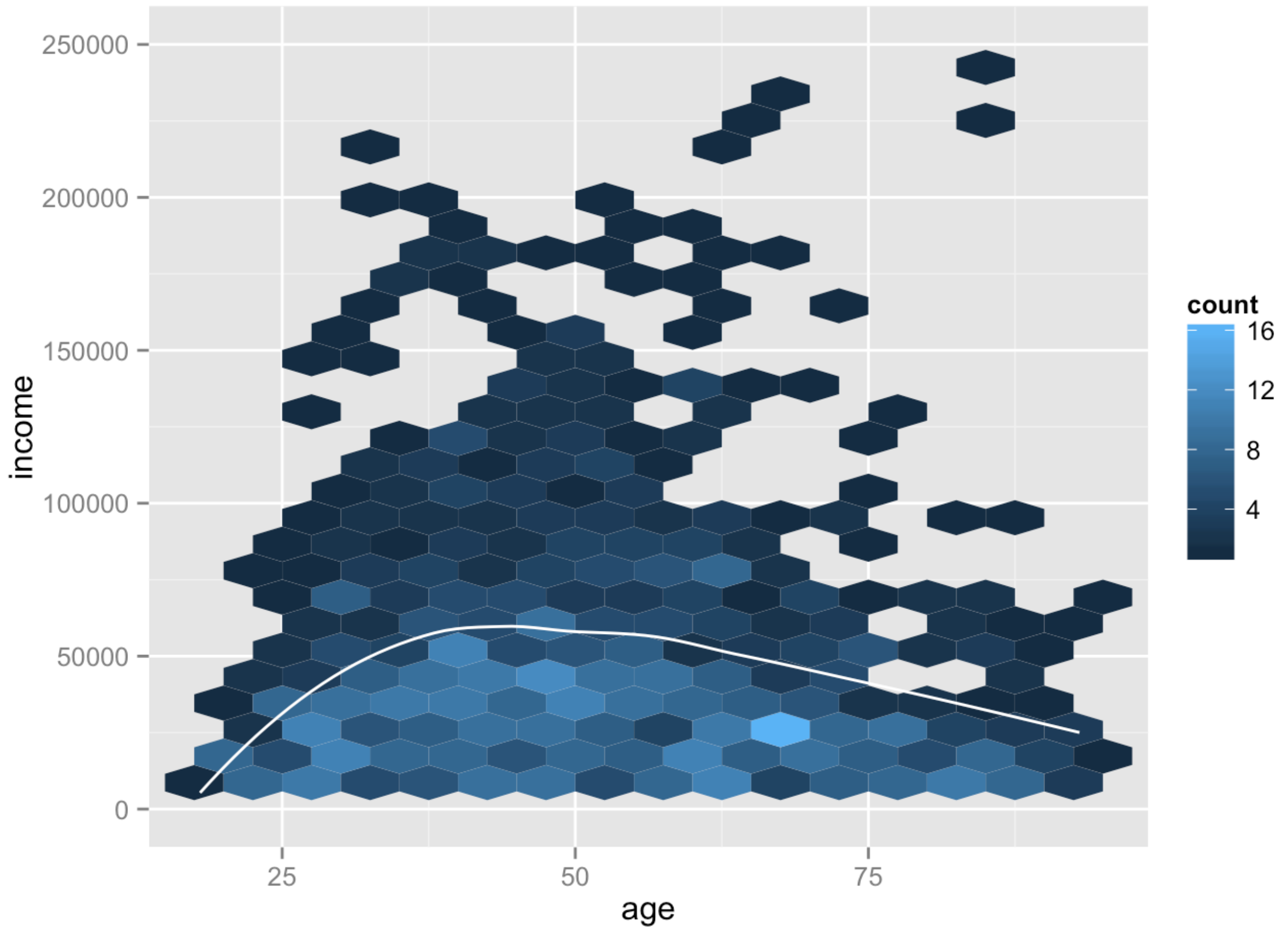
`hexbin` package

```
library(hexbin)
ggplot(custdata2, aes(x=age, y=income)) +
  geom_hex(binwidth=c(5, 10000)) +
  geom_smooth(colour="white", se=F) +
  ylim(0, 250000)
```

```
## Warning: Removed 25 rows containing missing values (stat_hexbin).
```

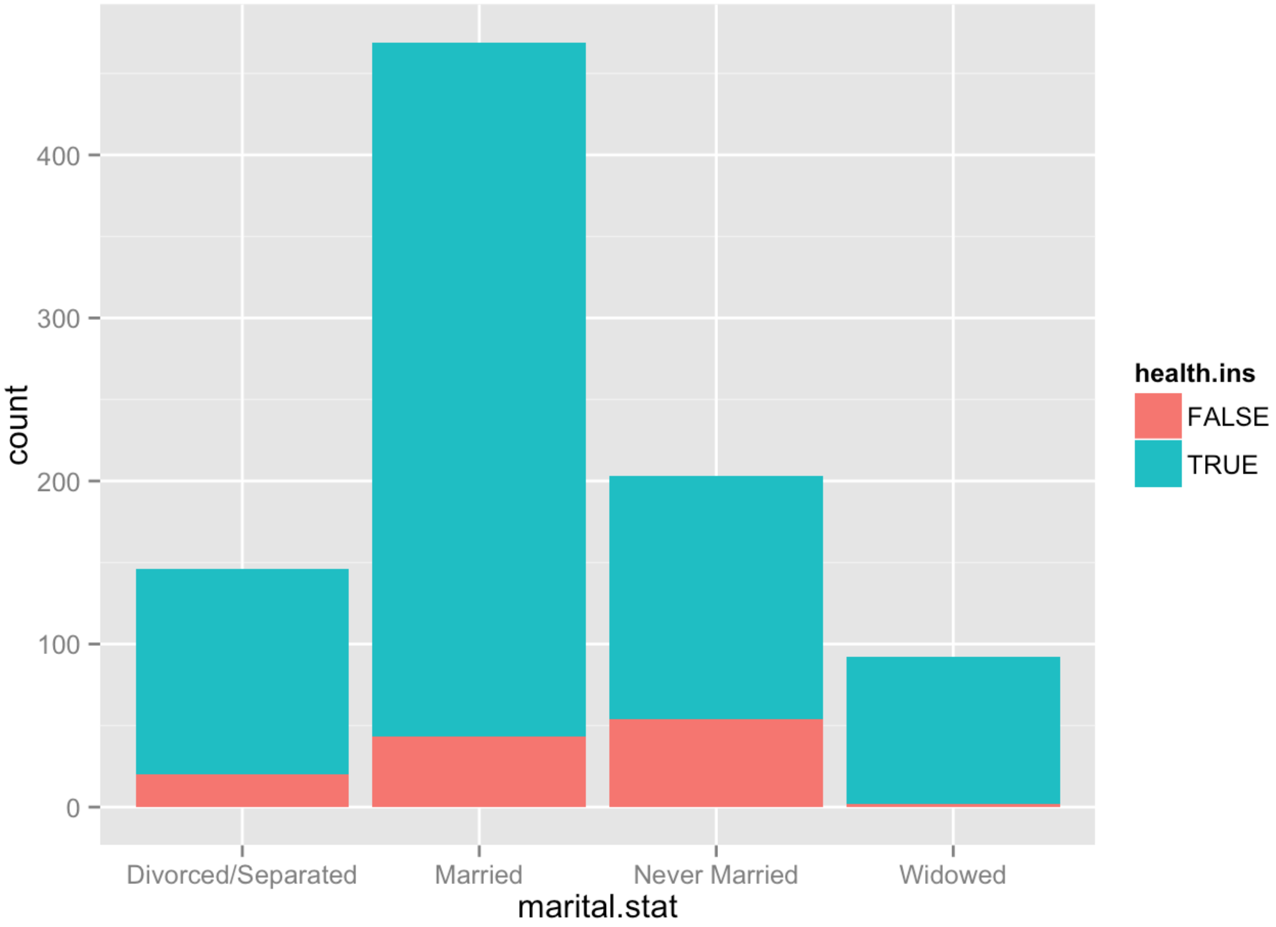
```
## geom_smooth: method="auto" and size of largest group is <1000, so using loes
s. Use 'method = x' to change the smoothing method.
```

```
## Warning: Removed 25 rows containing missing values (stat_smooth).
```



Bar Charts for Two Categorical Variables

```
ggplot(custdata2, aes(x=marital.stat, fill=health.ins)) + geom_bar()
```



- `table` 로 정리하고, data frame으로 만들어 작업하는데 있어서 한 가지 주의사항은 다음과 같이 `with()` 를 사용하여 `table` 로 만들어야 변수명을 그대로 사용할 수 있다는 점임.

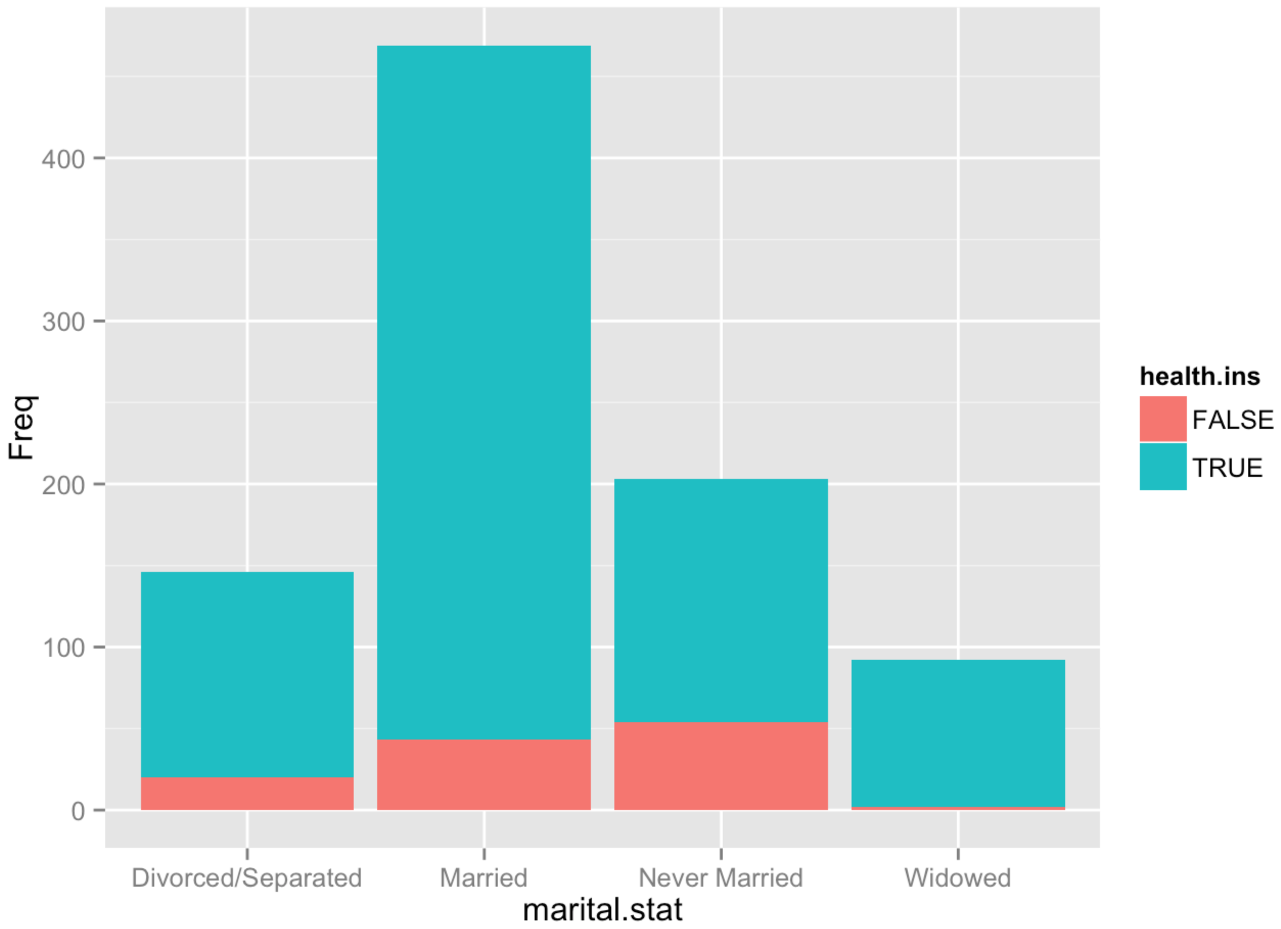
```
(tbl.mh <- with(custdata2, table(marital.stat, health.ins)))
```

```
##                       health.ins
## marital.stat           FALSE TRUE
##   Divorced/Separated      20  126
##   Married                 43  426
##   Never Married           54  149
##   Widowed                  2   90
```

```
(tbl.mh.df <- data.frame(tbl.mh))
```

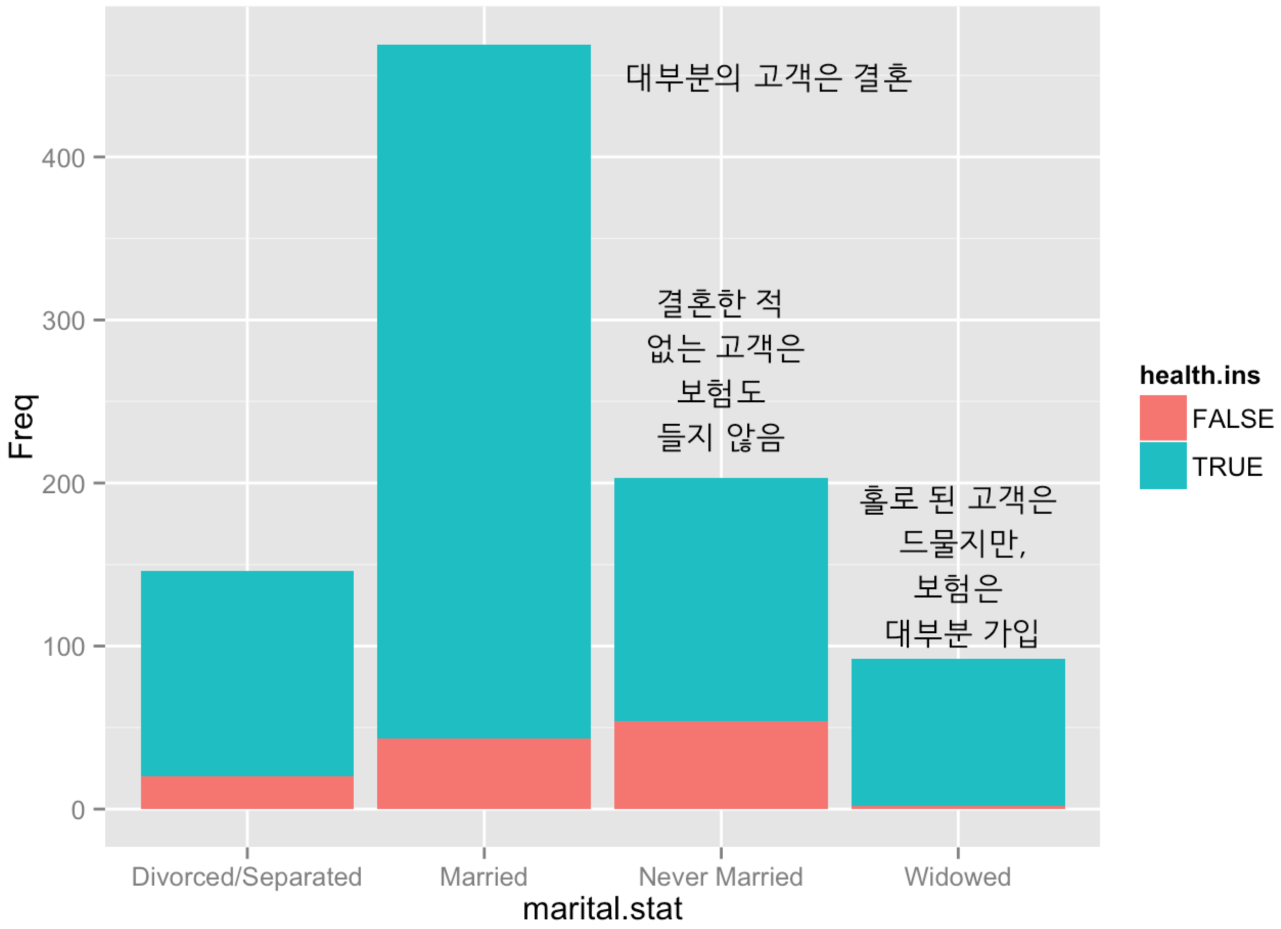
```
##         marital.stat health.ins Freq
## 1 Divorced/Separated      FALSE   20
## 2            Married      FALSE   43
## 3      Never Married      FALSE   54
## 4            Widowed      FALSE    2
## 5 Divorced/Separated       TRUE  126
## 6            Married       TRUE  426
## 7      Never Married       TRUE  149
## 8            Widowed       TRUE   90
```

```
(g.mh <- ggplot(tbl.mh.df, aes(x=marital.stat, y=Freq, fill=health.ins)) + geo
m_bar(stat="identity"))
```



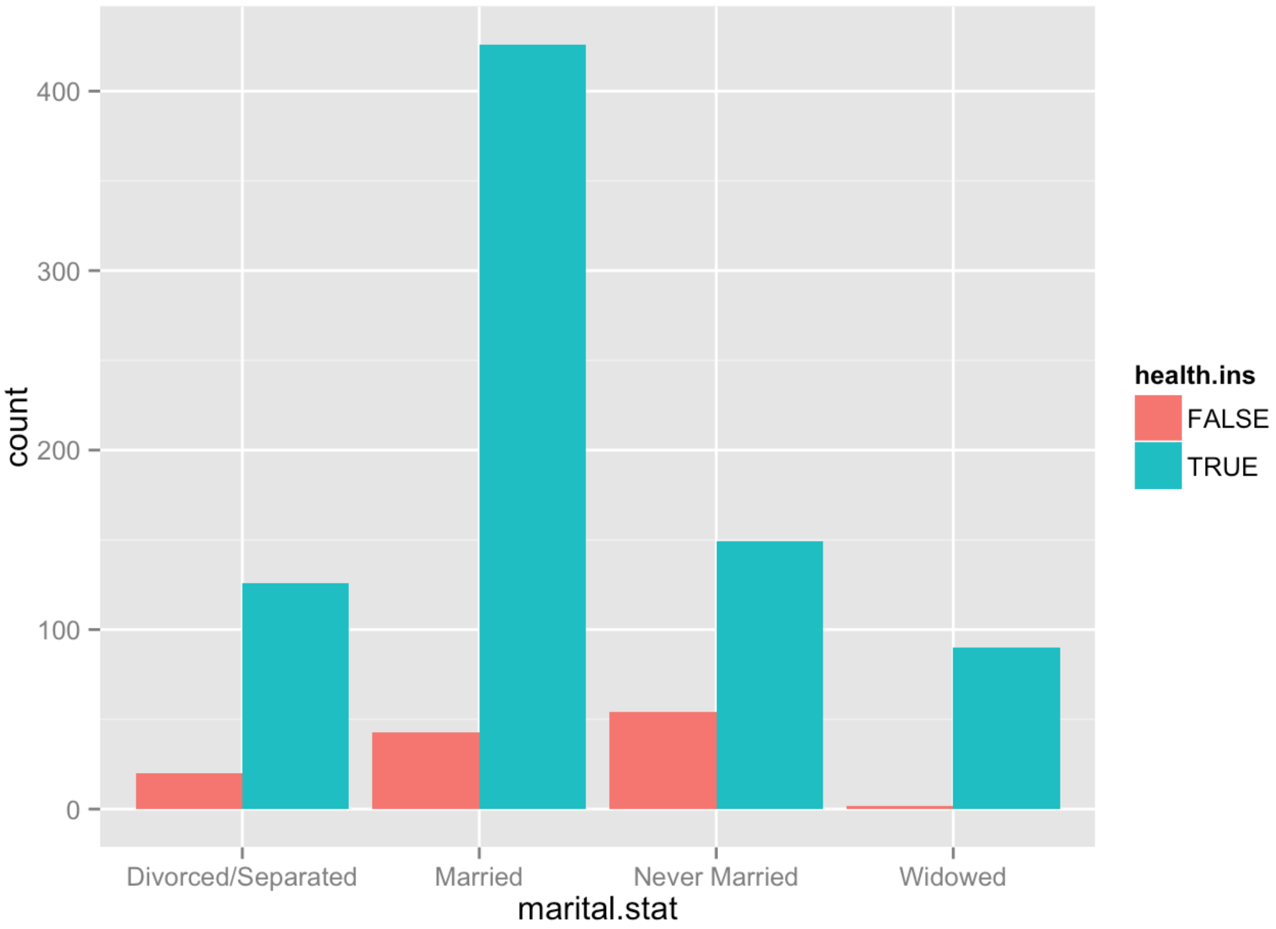
- 몇 가지 설명을 덧붙인다면,

```
g.mh + annotate("text", x=3.2, y=450, label="대부분의 고객은 결혼", family="HCR Dotu
m LVT", size=4) +
  annotate("text", x=3, y=270, label="결혼한 적\n 없는 고객은\n보험도\n들지 않음", famil
y="HCR Dotum LVT", size=4) +
  annotate("text", x=4, y=150, label="홀로 된 고객은\n 드물지만,\n보험은\n 대부분 가입",
family="HCR Dotum LVT", size=4)
```



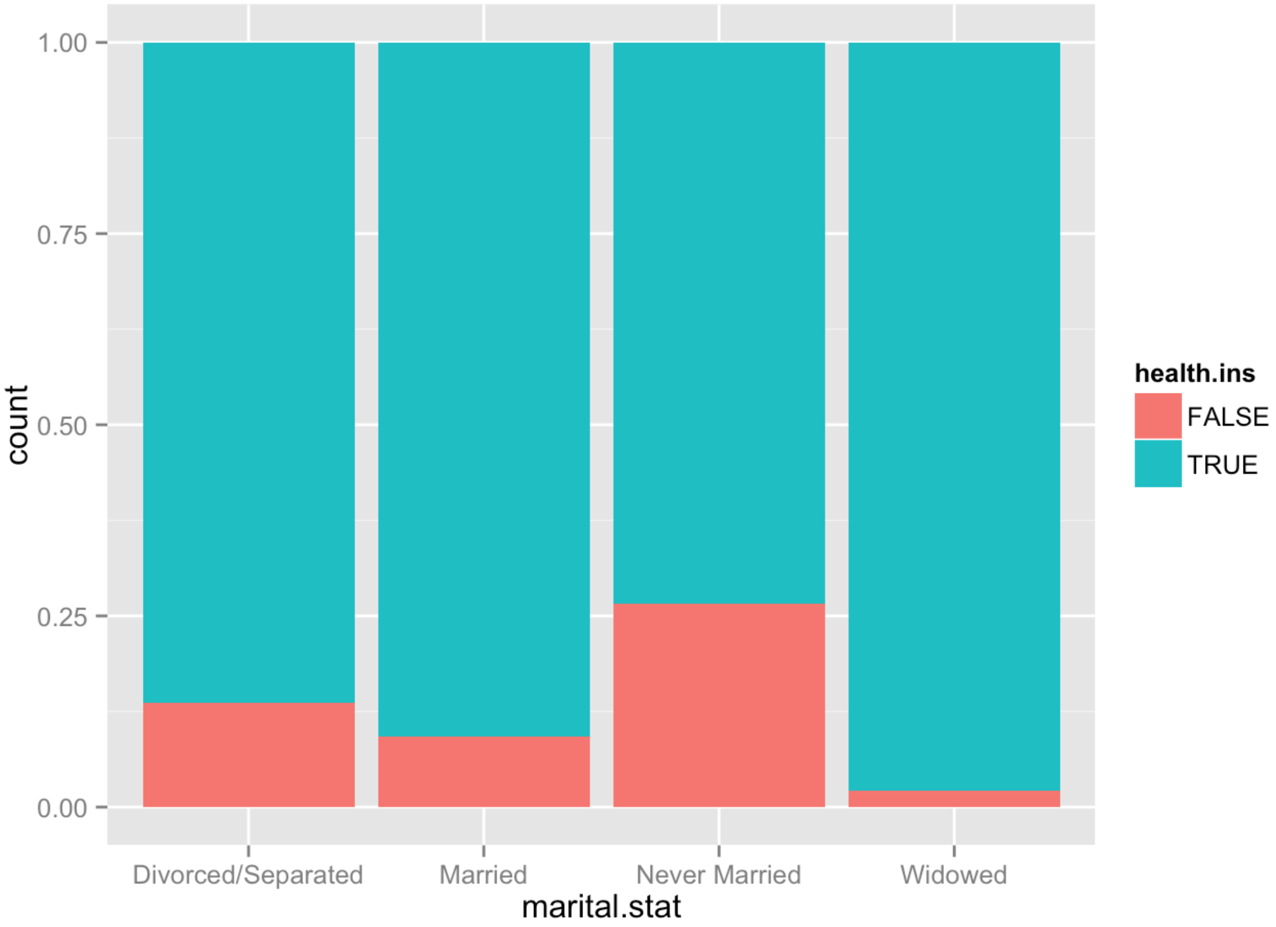
- `position="dodge"` 를 적용하면,

```
ggplot(custdata2, aes(x=marital.stat, fill=health.ins)) + geom_bar(position="do
dge")
```



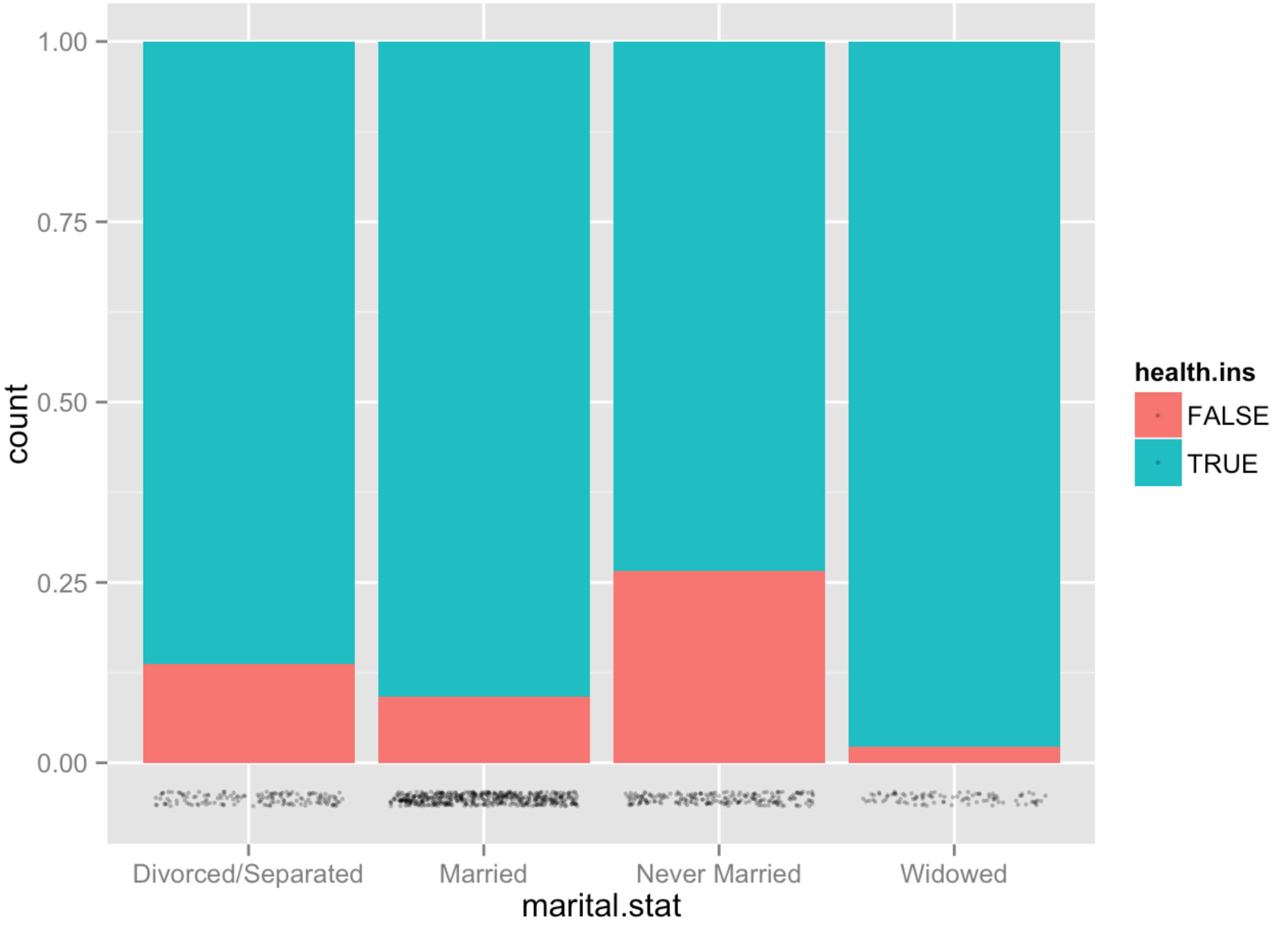
- `postion="fill"` 를 적용하면,

```
ggplot(custdata2, aes(x=marital.stat, fill=health.ins)) + geom_bar(position="fi
ll")
```



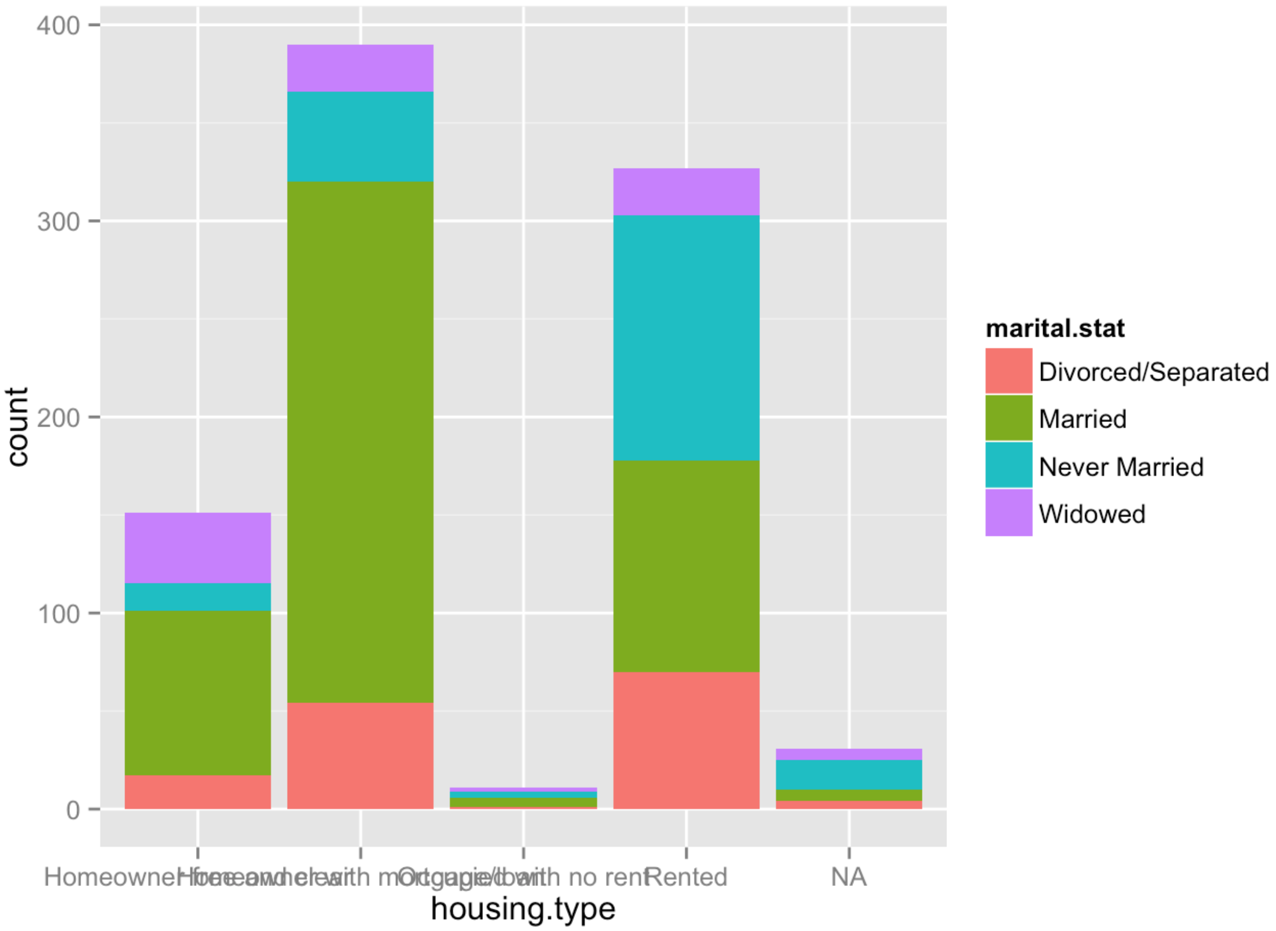
- rug 를 설정하면,

```
ggplot(custdata2, aes(x=marital.stat, fill=health.ins)) + geom_bar(position="fi
ll") +
  geom_point(aes(y=-0.05), size=0.75, alpha=0.3, position=position_jitter(h=0.0
1))
```

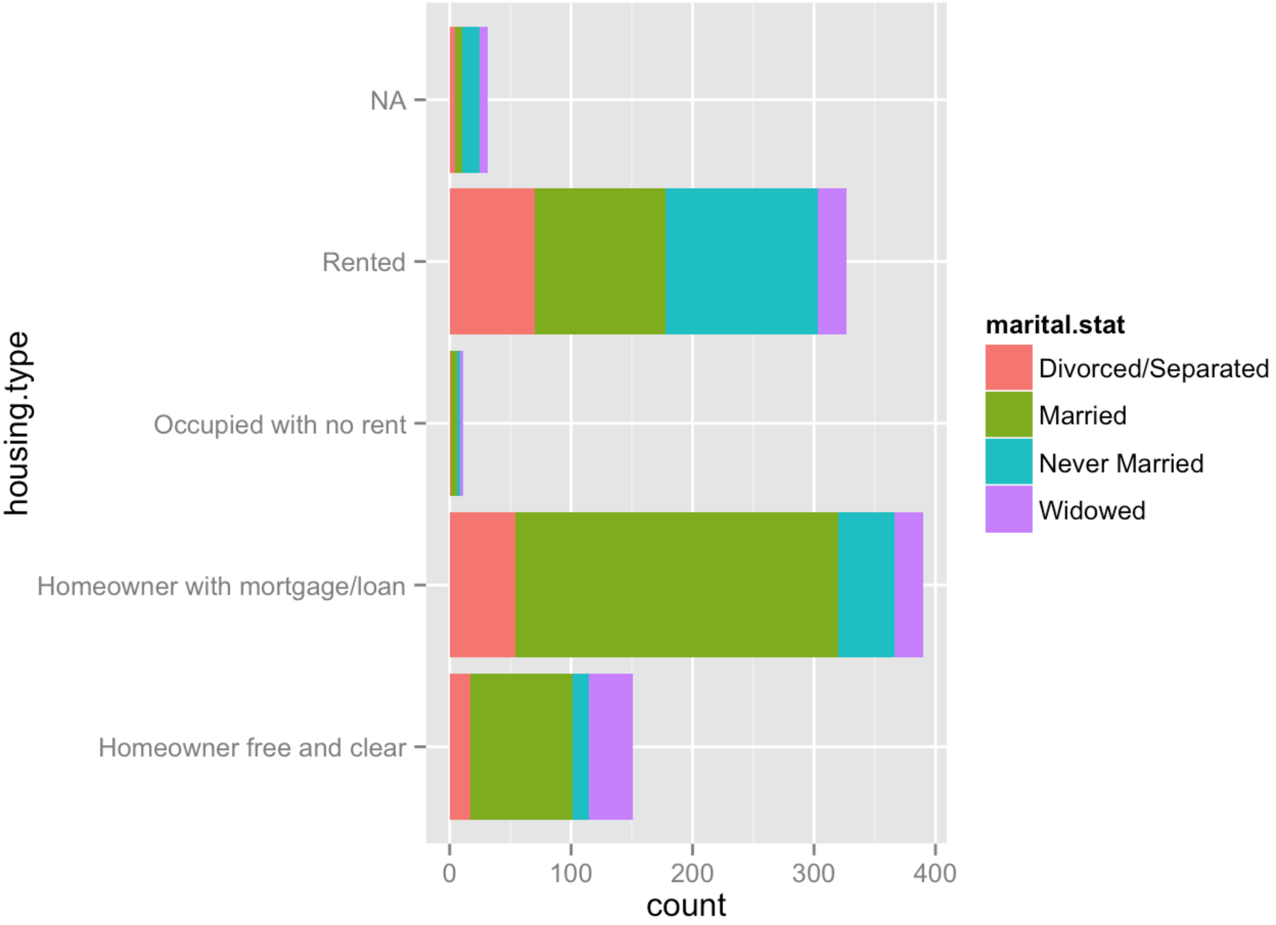


- Listing 3.17

```
(g.hm <- ggplot(custdata2, aes(x=housing.type, fill=marital.stat)) + geom_ba
r())
```



```
g.hm + coord_flip()
```



- 보기 좋게 다시 그리면,

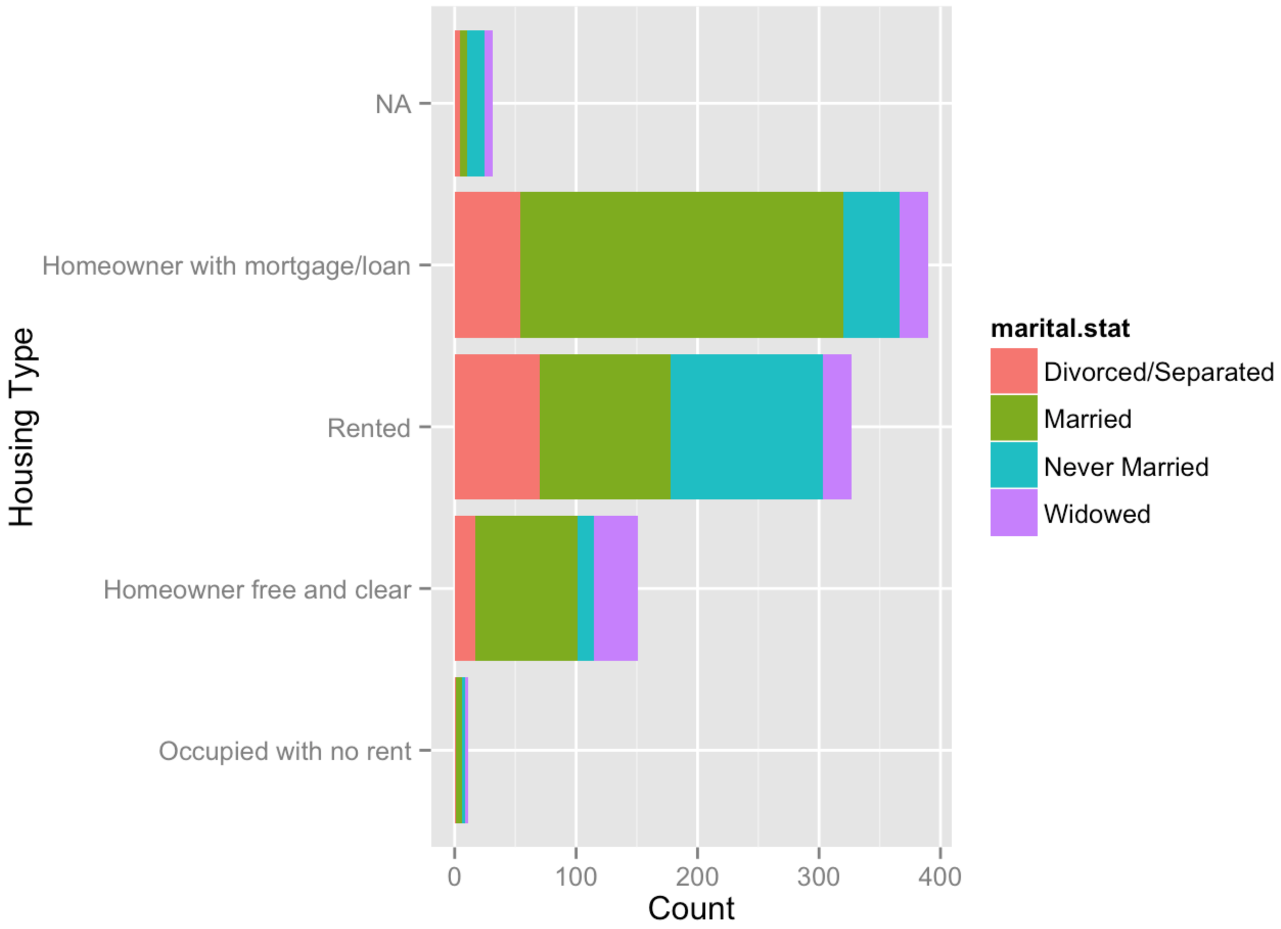
```
(tbl.hm <- with(custdata2, table(housing.type, marital.stat, useNA="ifany")))
```

```
##                               marital.stat
## housing.type                   Divorced/Separated Married Never Married
##   Homeowner free and clear                     17      84            14
##   Homeowner with mortgage/loan                 54     266            46
##   Occupied with no rent                         1       5             3
##   Rented                                       70     108           125
##   <NA>                                          4       6            15
##                               marital.stat
## housing.type                   Widowed
##   Homeowner free and clear          36
##   Homeowner with mortgage/loan      24
##   Occupied with no rent              2
##   Rented                            24
##   <NA>                               6
```

```
(tbl.hm.df <- data.frame(tbl.hm))
```

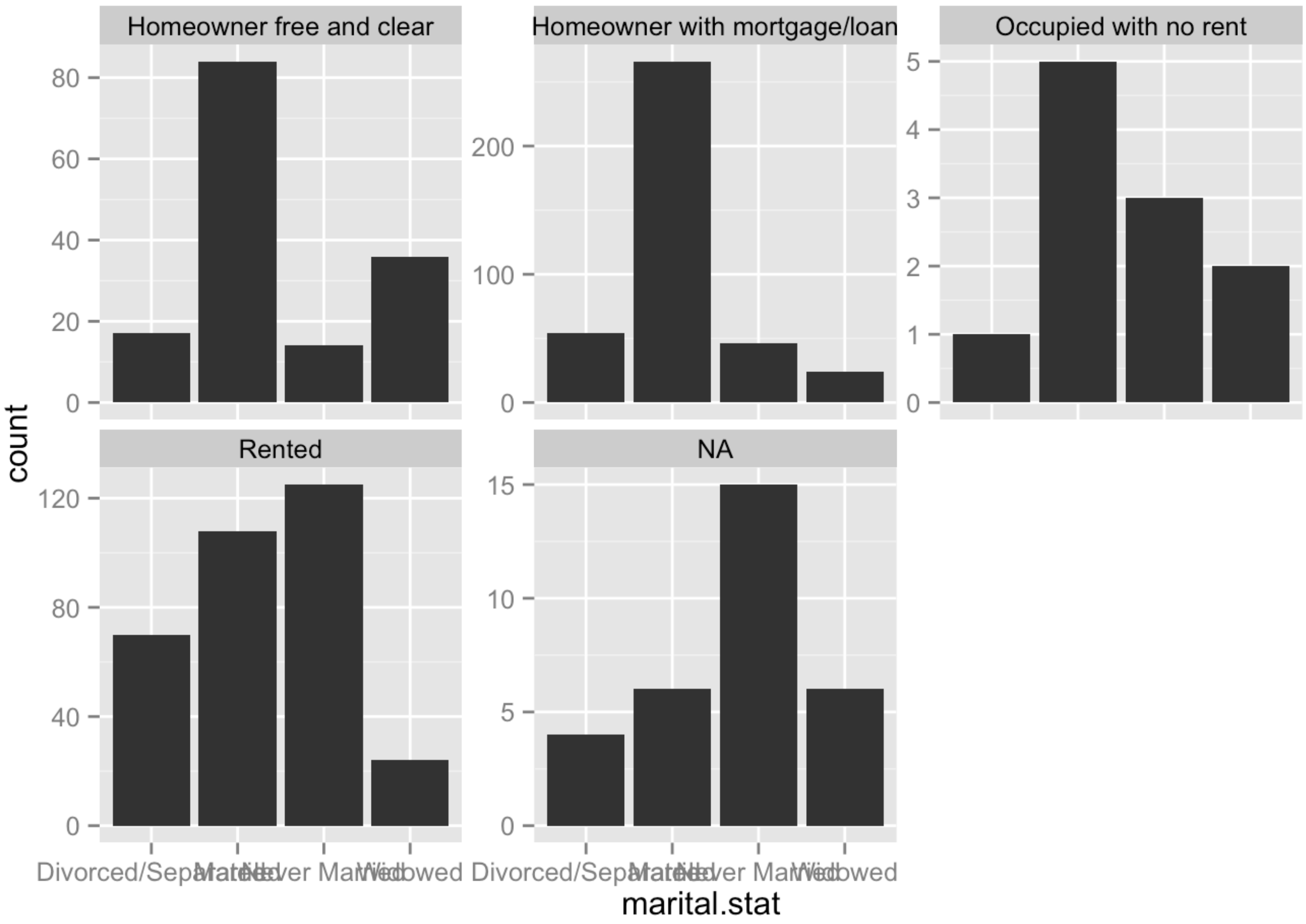
```
##                     housing.type       marital.stat Freq
## 1      Homeowner free and clear Divorced/Separated   17
## 2  Homeowner with mortgage/loan Divorced/Separated   54
## 3         Occupied with no rent Divorced/Separated    1
## 4                        Rented Divorced/Separated   70
## 5                          <NA> Divorced/Separated    4
## 6      Homeowner free and clear            Married   84
## 7  Homeowner with mortgage/loan            Married  266
## 8         Occupied with no rent            Married    5
## 9                        Rented            Married  108
## 10                         <NA>            Married    6
## 11     Homeowner free and clear      Never Married   14
## 12 Homeowner with mortgage/loan      Never Married   46
## 13        Occupied with no rent      Never Married    3
## 14                       Rented      Never Married  125
## 15                         <NA>      Never Married   15
## 16     Homeowner free and clear            Widowed   36
## 17 Homeowner with mortgage/loan            Widowed   24
## 18        Occupied with no rent            Widowed    2
## 19                       Rented            Widowed   24
## 20                         <NA>            Widowed    6
```

```
ggplot(tbl.hm.df, aes(x=reorder(housing.type, Freq), y=Freq, fill=marital.sta
t)) +
  geom_bar(stat="identity") +
  coord_flip() +
  xlab("Housing Type") + ylab("Count")
```

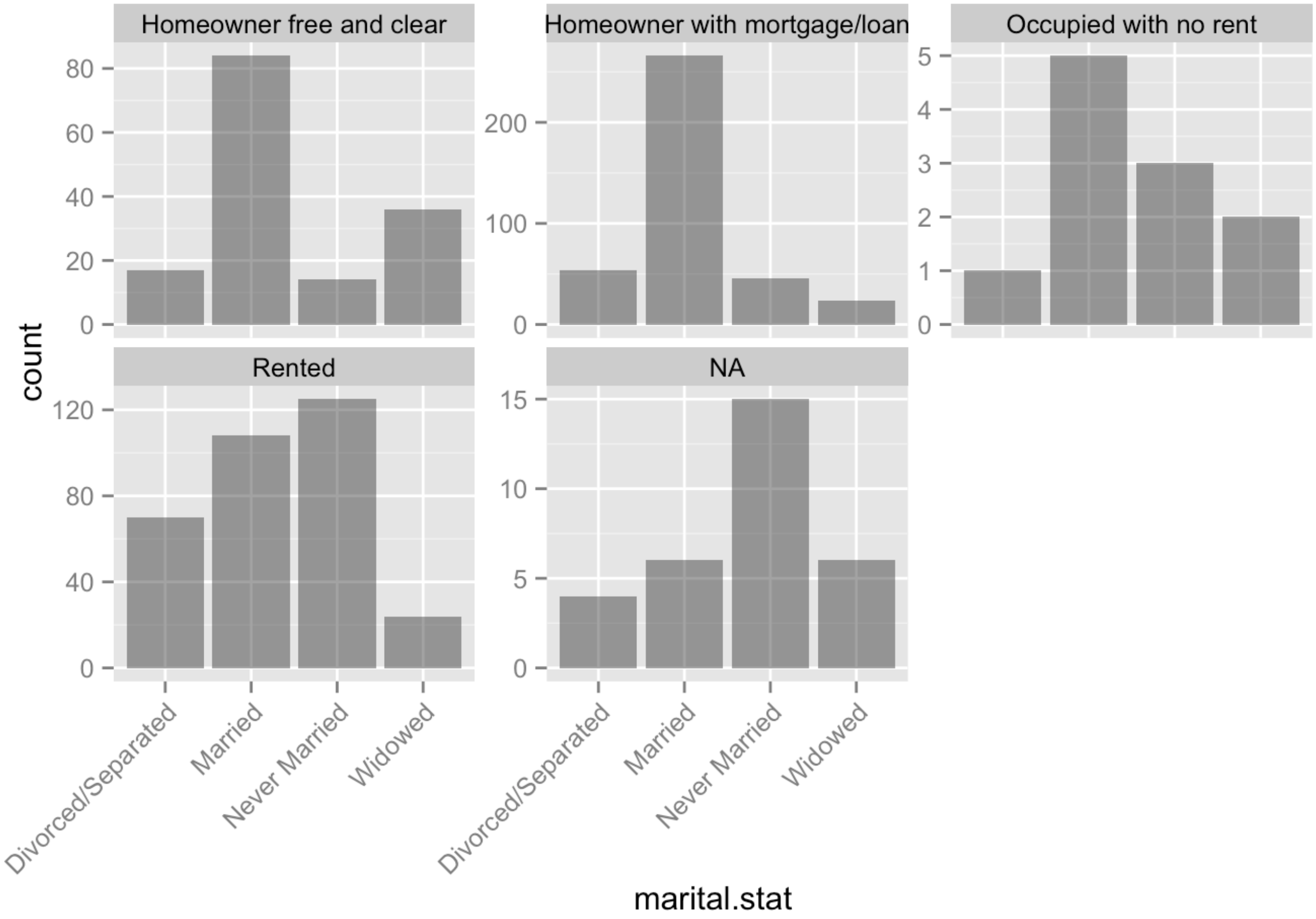


- `facet_wrap()` 을 활용하면,

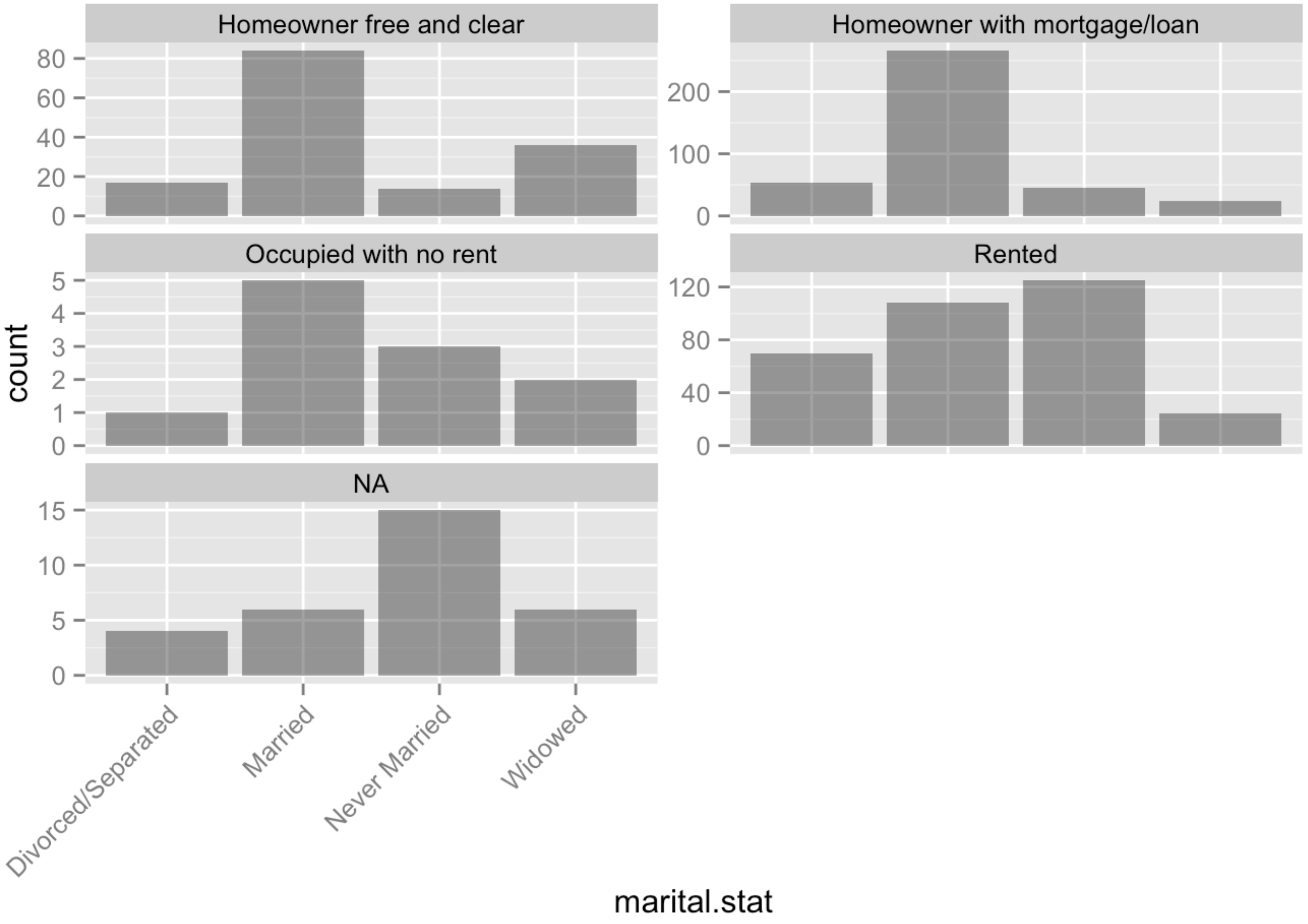
```
ggplot(custdata2, aes(x=marital.stat)) + geom_bar(position="dodge") +
  facet_wrap(~housing.type, scales="free_y")
```



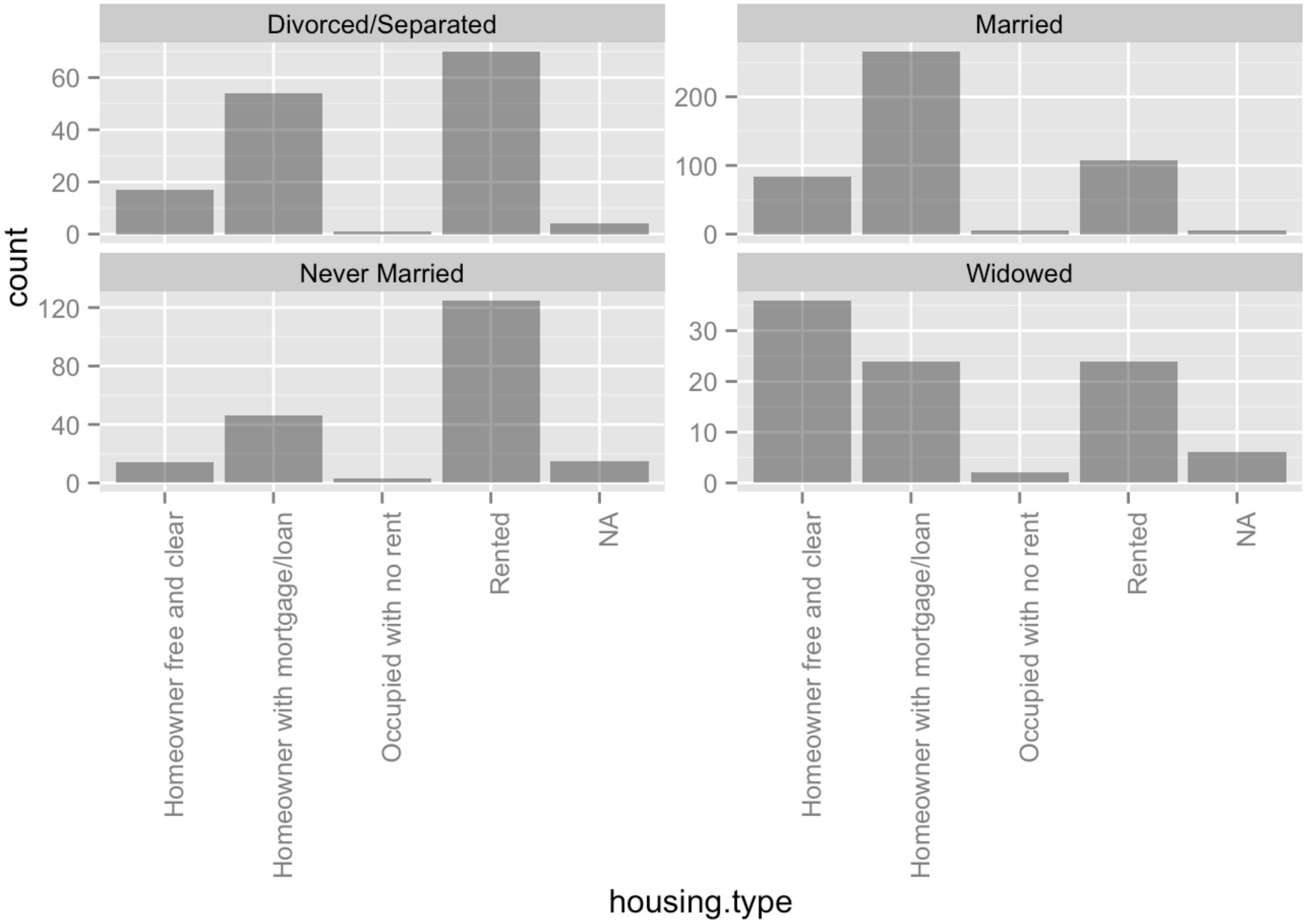
```
ggplot(custdata2, aes(x=marital.stat)) + geom_bar(position="dodge", alpha=0.5)
+
  facet_wrap(~housing.type, scales="free_y") +
  theme(axis.text.x = element_text(angle=45, hjust=1))
```



```
ggplot(custdata2, aes(x=marital.stat)) + geom_bar(position="dodge", alpha=0.5)
+
  facet_wrap(~housing.type, scales="free_y", ncol=2) +
  theme(axis.text.x = element_text(angle=45, hjust=1))
```

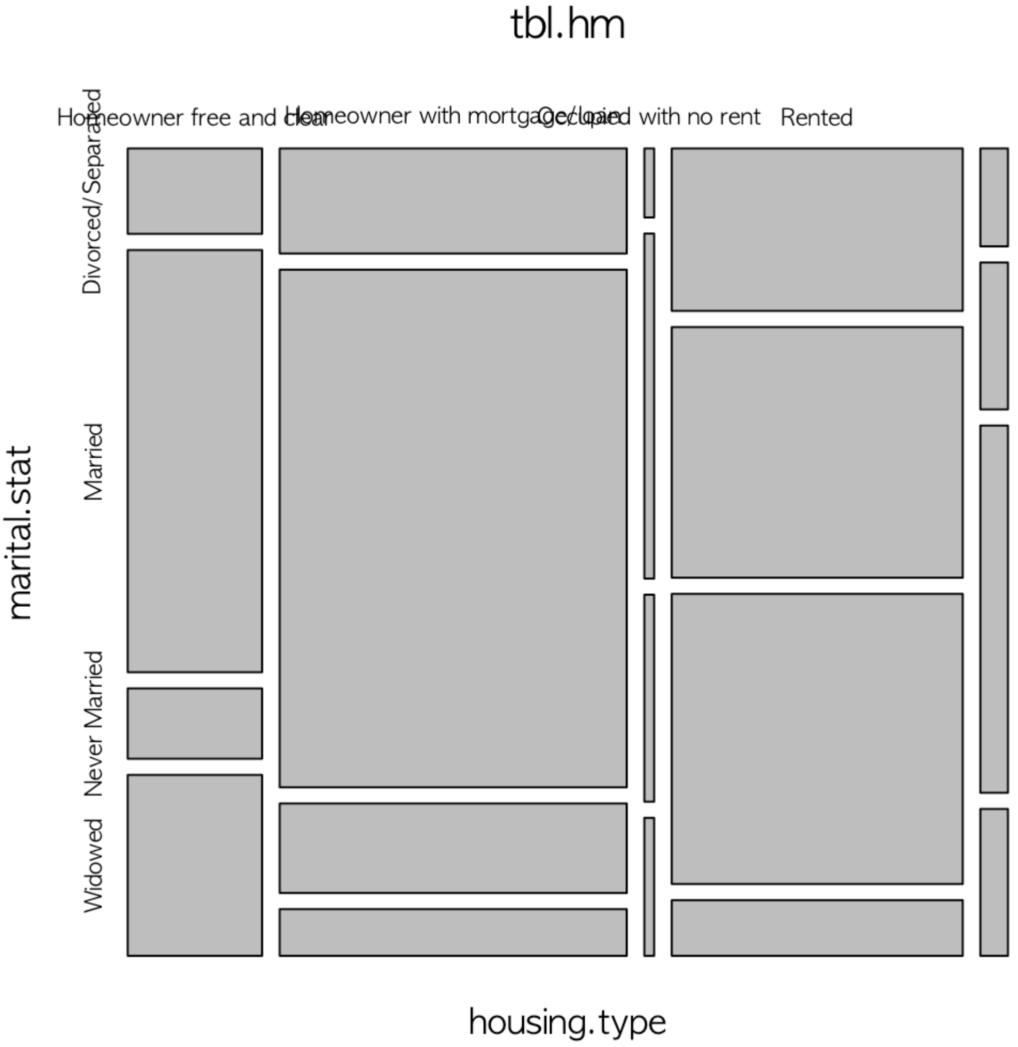


```
ggplot(custdata2, aes(x=housing.type)) + geom_bar(position="dodge", alpha=0.5)
+
  facet_wrap(~marital.stat, scales="free_y", ncol=2) +
  theme(axis.text.x = element_text(angle=90, hjust=1))
```

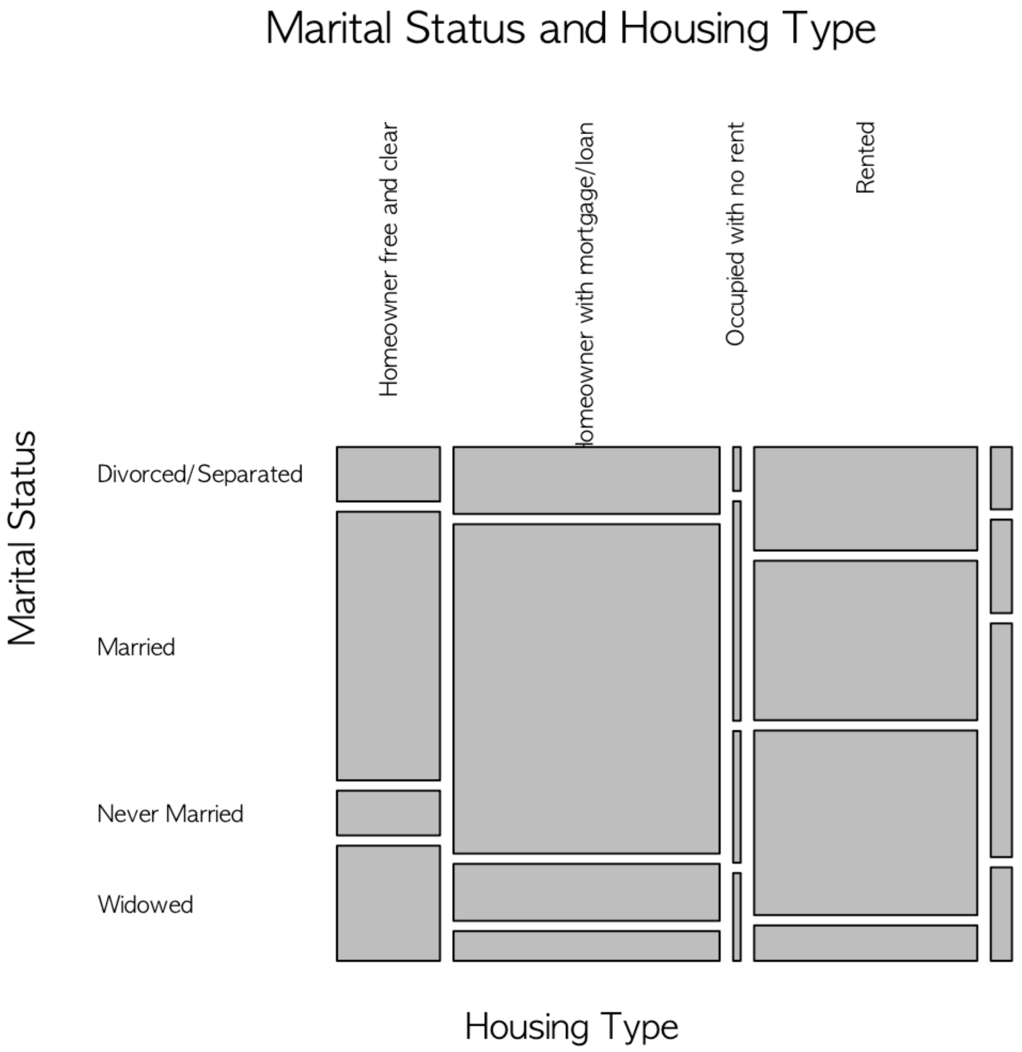


- `mosaicplot()` 을 사용하면,

```
mosaicplot(tbl.hm)
```

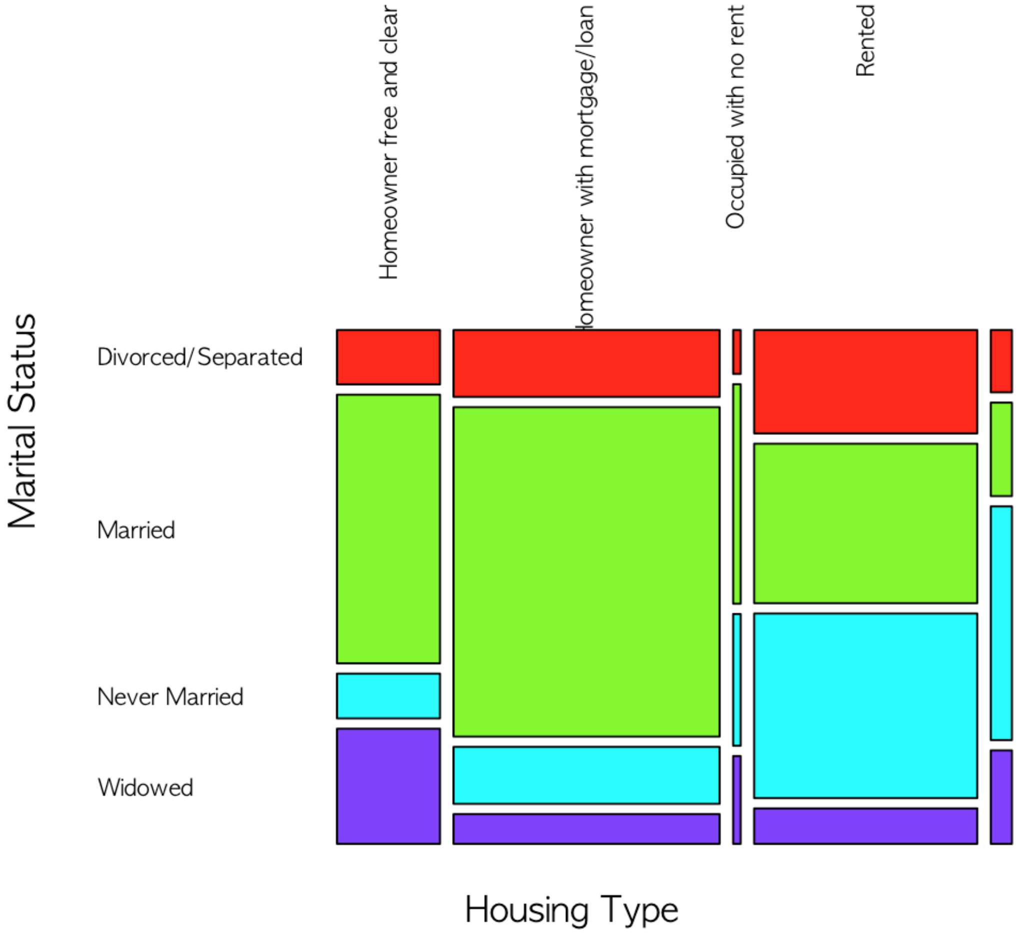


```
mosaicplot(tbl.hm, main="Marital Status and Housing Type", xlab="Housing Type",
ylab="Marital Status", las=2)
```



```
mosaicplot(tbl.hm, main="Marital Status and Housing Type", xlab="Housing Type",
ylab="Marital Status", las=2, color=rainbow(4))
```

Marital Status and Housing Type


Homeowner free and clear
Homeowner with mortgage/loan
Occupied with no rent
Rented
Marital Status
Divorced/Separated
Married
Never Married
Widowed
Housing Type